Summer 8-8-2023

Towards non-vascular fundus image analysis and disease detection

Saeid Motevali

Recommended Citation
Motevali, Saeid, "Towards non-vascular fundus image analysis and disease detection." Dissertation, Georgia State University, 2023.
doi: https://doi.org/10.57709/35838868

Towards non-vascular fundus image analysis and disease detection

by

Saeid Motevali

Under the Direction of Rolando Estrada, Ph.D.

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

in the College of Arts and Sciences

Georgia State University

2023

ABSTRACT

Assessment of retinal fundus image is very informative and preventive in early ocular disease detection. This non-invasive assessment of fundus images also helps in the early diagnosis of vascular diseases. This unique combination help in the early diagnosis of diseases. Applying image enhancement techniques with advanced Deep learning techniques helps to overcome such a challenging problem. Most Deep learning models give a diagnosis without attention to underlying pathological abnormalities. In this thesis, we tried to solve the problem in the same way as ophthalmologists and experts in the field approach the problem. We created models that can detect an Optic disc, Optic cup, and vascular regions in the image. This work can be integrated into any ocular disease detection, such as glaucoma, and vascular disease detection, such as diabetes. Extensive work is applied for better sampling when all models were suffering from a lack of data in the medical imaging field. The entire work on the retinal fundus image was in 2d images. In the extension of this work, we applied our knowledge to 3d MRI-Brain images. We attempt to predict attention scores in children, which is a big factor in the detection of kids with ADHD. But both work on fundus images and brain MRI images are under the umbrella of medical imaging. We believe this advancement in this line of research can be very valuable for future researchers in the area of automated medical imaging, especially in automated retinal disease diagnosis.

INDEX WORDS: Fundus image analysis, Optic disc detection, Optic cup detection, Vessel segmentation, Brain-MRI images, Automated disease diagnosis, Deep learning

Towards non-vascular fundus image analysis and disease detection

by

Saeid Motevali

Committee Chair: Rolando Estrada

Committee: Jingyu Liu

Rajshekhar Sunderraman

Jocelyn Desbiens

Electronic Version Approved:

Office of Graduate Studies

College of Arts and Sciences

Georgia State University

August 2023

DEDICATION

I like to dedicate this work to my best friend, my beautiful and smart wife Samaneh Bitarafan, who supported me all along in this long journey. To my lovely kids, Benjamin and Diana, who I hope to follow the same path and educated themselves to benefit themselves and society. To my parents, Azam and Alireza who raised me and made education a big part of my life. Also to my brothers and sisters. I also want to recognize my friend and colleagues who helped me through this work.

ACKNOWLEDGEMENTS

Special thanks to my knowledgeable advisor Dr. Rolando Estrada, who is a great human being, and I wish him nothing but success in his future journey. I like to thank him for this great opportunity. I also like to thank Dr. Raj Sunderraman, who was always there when we needed him the most. I want to thank the department of computer science at GSU and TReNDS Center team Dr. Liu and Dr. Vince Calhoun for this great research opportunity. Special thanks to Dr. Jocelyn for helping us understand the topics better. I also want to thank my first advisor in graduate school, Dr. Khalil(Ali) Shujaee who believed in me and gave me the opportunity to start this journey. I wish all of you the best.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS **v**

LIST OF TABLES **ix**

LIST OF FIGURES **x**

1 INTRODUCTION **1**

2 Fundus Optic Disc Segmentation **2**

2.1 Introduction 3

2.2 Methodology 7

2.2.1 *U-net architecture* 7

2.2.2 *Disc Centered Patch Augmentation* 9

2.2.3 *Stochastic cost functions* 11

2.3 Experiments and Results 12

2.4 Discussion 17

2.5 Conclusions and Future work 18

3 Fundus Optic Cup Segmentation **20**

3.1 Introduction 20

3.2 Methodology 21
3.2.1 *Cup Centered Patch Augmentation* 22
3.2.2 *Stochastic cost functions* 23
3.3 Results and future work 23

4 Attention Score in Brain MRI **25**
4.1 Introduction 25
4.2 ABCD data 26
4.3 Data Preparation 27
4.4 Models 28
4.5 Age Prediction 31
4.6 Gender prediction 32
4.7 Attention score Classification 34
4.8 Attention Score Prediction 35
4.9 Transfer learning 36
4.10 Conclusion 40

5 Vessele Tree Topology **41**
5.1 Introduction 41
5.2 Related Work 44
5.2.1 *Graph topology* 45
5.2.2 *Segmentation* 46
5.2.3 *Classification* 47
5.3 Methodology 48

5.3.1 *Topology Extraction with Multilevel Skeletonization* 48

5.4 Discussion 50

5.5 Conclusion and Future Work 51

6 CONCLUSION **52**

References **53**

LIST OF TABLES

2.1 Dataset details 8

2.2 Optic disc result 14

2.3 Optic disc run time 17

3.1 Dataset cup details 21

3.2 Optic cup result 24

LIST OF FIGURES

2.1 Center of mass shift 10

2.2 Disc segmentation results of different datasets 19

3.1 Cup segmentation results of DRISTI datase 23

4.1 Brain MRI visualization 26

4.2 VGG-16 detailed architecture 29

4.3 VGG-16 convolution layers 29

5.1 Flowchart of the whole vessel probability map pipeline 42

5.2 Original image, manual vessel segmentation, vessel prior 45

1 | INTRODUCTION

Fundus retinal images are the only available source to see the structure of vessels, which is non-invasive. It is very easy to capture such an image with a fundus camera, and it is cost-effective. These fundus images have been very well studied in the last two decades (Abràmoff et al., 2010). The captured images by fundus cameras are of different quality and resolutions based on the quality of the fundus camera. These images are also in different colors, lighting, and size. The fundus images are heterogeneous, and that makes it more challenging to create a model. As is often observed, the evaluation of the patient's eye condition may vary depending on the ophthalmologist's opinion. It is also a challenge for an automated system to predict and diagnose in the environment that they did not train on. In the fundus image research for this thesis, we overcome that issue with the help of Deep learning techniques. Our model trained on different domains regardless of the patient's age, gender, and health condition. That makes the model very robust and flexible(Gretton et al., 2009). In another line of research in medical imaging, we worked on the attention score prediction for brain-MRI 3D images. We attempt to predict the attention score solely by using brain-MRI images on the ABCD dataset. The ABCD datset is avalable at `https://nda.nih.gov/abcd/`

2 | Fundus Optic Disc Segmentation

The optic disc is a crucial diagnostic feature in the eye since changes to its physiognomy are correlated with the severity of various ocular and cardiovascular diseases. While identifying the bulk of the optic disc in a color fundus image is straightforward, accurately segmenting its boundary at the pixel level is very challenging. In this work, we propose disc-centered patch augmentation (DCPA)—a simple yet novel training scheme for deep neural networks—to address this problem. DCPA achieves state-of-the-art results on full-size images even when using small neural networks, specifically a U-Net with only 7 million parameters as opposed to the original 31 million. In DCPA, we restrict the training data to patches that fully contain the optic nerve. In addition, we also train the network using dynamic cost functions to increase its robustness. We tested DCPA-trained networks on five retinal datasets: DRISTI, DRIONS-DB, DRIVE, AV-WIDE, and CHASE-DB. The first two had available optic disc ground truth, and we manually estimated the ground truth for the latter three. Our approach achieved state-of-the-art F1 and IOU results on four datasets (95% F1, 91% IOU on DRISTI; 92% F1, 84% IOU on DRIVE; 83% F1, 71% IOU on AV-WIDE; 83% F1, 71% IOU on CHASEDB) and competitive results on the fifth (95% F1, 91% IOU on DRIONS-DB), confirming its generality. Our open-source code and ground-truth annotations are available at `https://github.com/saeidmotevali/fundusdisk`.

2.1 Introduction

The optic disc is the region where ganglion cell axons and blood vessels exit the retina. The optic disc is a crucial diagnostic structure in the eye since changes to its physiognomy are correlated with the severity of various diseases, including glaucoma Varma et al. (1992), idiopathic intracranial hypertension (IIH) Digre et al. (2009), coronary heart disease Rochtchina et al. (2007), and atherosclerosis Ikram et al. (2004). For example, the cup-to-disc ratio (CDR), which measures the ratio of the disc to its central depression (cup), is widely used to diagnose glaucoma Walter and Klein (2001); Youssif et al. (2008); Zhu et al. (2010). Glaucomatous eyes show more pronounced cupping due to an increase in intracranial pressure, which causes flowing nutrients to pass through damaged axons and induce swelling of the optic disc Wang et al. (2016). Optic disc swelling can also be a sign of a brain tumor or a minor nerve stroke.

Many methods have been proposed for automatically segmenting the optic disc in color fundus images. In general, the bulk of the optic disc is easy to detect since it is brighter and yellower than the rest of the retina Zhu et al. (2010). However, automatically determining the disc's *boundary* at the pixel level is challenging due to multiple factors. First, imaging artifacts such as motion or blur can obscure the edges of the disc. Second, retinal diseases can make this boundary more irregular, making it harder for an example-trained system (e.g., a deep neural network) to accurately trace it on non-healthy eyes. Also, existing datasets for this problem are small and relatively rare, even compared to other retinal segmentation problems. Finally, an additional challenge of optic disc segmentation is that the percentage of positive pixels is extremely low Cheng et al. (2011); Maninis et al. (2016); Mohan et al.

(2018). Specifically, the optic disc only constituted an average of 3.46% of the image across the five datasets in our experiments (see Tbl. 2.1 for details).

To address these issues, we propose a novel scheme for training deep neural networks called *disc center patch augmentation* (DCPA). Our approach is based on the U-net architecture Ronneberger et al. (2015), the current state-of-the-art deep architecture for medical segmentation problems. Specifically, in DCPA we restrict the training data to only include the region of the fundus image where the optic disc is located, improving the robustness of the trained model. In more detail, one can use a U-net-style network with a fixed input size for images of arbitrary size by splitting the input into *patches*. These patches are fed to the neural network independently and then stitched together to form the final output. This patch-based approach is common in medical image segmentation. However, it is challenging to apply it to optic disc segmentation because the optic disc is significantly smaller than the fundus image as a whole. As such, the disc may be only present in a few of the patches. Patches without any positive pixels are problematic because (1) they do not contain any useful optic disc features, and (2) they can induce the network to become oversensitive to noise in the image. Thus, in DPCA, we ensure that all the patches that we feed to the network during training contain the entire optic disc. We vary the position of the optic disc across patches to prevent location-based bias. Our implementation works with the original images (i.e., without resizing). Thus, it is compatible with downstream bio-markers for ocular disease diagnosis, such as the optic cup-to-disc ratio.

In addition to DCPA, we allow make neural network training more robust by applying dynamic cost functions Khanal and Estrada (2020a) to this problem. Stochastic penalties allow a network to settle on an optimum that is more robust to ambiguous pixels (e.g., those

at anatomical boundaries) than conventional training. In particular, a network trained with stochastic weights achieved state-of-the-art results in numerous retinal vessel segmentation datasets Khanal and Estrada (2020a).

Training a deep neural network on medical images is very resource intensive due to their large size. In addition, it is not desirable to reduce the image size prior to processing because it might introduce unwanted pixel-level artifacts and/or loss of crucial information in the up/down-sampling process, both of which can lead to misdiagnoses. Fortunately, as we detail in Sec. 3.3, our DCPA strategy allows us to achieve comparable results on OD segmentation using a much smaller network (at least 4.5 times smaller than a conventional U-net). We experimentally validated our approach across five retinal datasets—DRISTI, DRIVE, DRIONS-DB, AV-WIDE, and CHASE-DB—achieving state-of-the-art results on four of them and competitive results on the fifth. In addition to our comparison against the state of the art, we also carried out ablation studies on the different components of our system to better understand their impact on performance. Overall, we determined that centered patches consistently improved results, while the benefit of dynamic weights was more dataset-specific.

On a color fundus image, the optic disc appears as a distinctive bright spot in the retina, located next to a dense, relatively dark cluster of vessels. Researchers have applied different techniques such as structural detection and convex hull Roychowdhury et al. (2016), active contour Claudia Kondermann (2007), and edge filter Cheng et al. (2011) to automatically segment this area. However, deep learning techniques are well known to learn low-to-high-level features that generalize significantly better than other machine learning methods. Mohan et al. (2018) used CNN architecture, Ronneberger et al. (2015) used U-Net and Maninis

et al. (2016) applied deep retinal image understanding for OD segmentation. Below, we review some of the main techniques proposed for this problem.

in Roychowdhury et al. (2016) applied morphological reconstruction and applied a circular structure element on the green channel of the fundus image. Then the bright region, which is adjacent to major blood vessels detected as the optic disc. In the next step, binary classification is applied to classify the bright region into OD and non-OD regions. The area with maximum vessel sum and solidity is considered the best candidate for OD. The other area is within a 1-disc diameter from the centroid of the best remaining OD candidate. After that, the convex hull containing all the candidate OD regions is considered, and the best-fit ellipse across the convex hull defines the segmentation OD boundary.

in Claudia Kondermann (2007) used information of contrast and texture in the image to detect the optic disc. The model uses a statistical-based method to detect the optic disc contour in fundus retina images. After the initial guess of the contour, the method segments the optic disc further by using the active contour model (ACM) or snakes. A snake is a deformable boundary around an object for which an image's external energy and the contour's internal energy shape the final contour detection. There are a number of parameters that affects the final contour shape. Any variation in these parameters has a massive effect on the final contour detection. Also, such parameters need to be tuned for each new dataset Pallawala et al. (2004).

Peripapillary atrophy elimination was used in Cheng et al. (2011) by to detect the optic disc. The elimination applied edge filtering, constraint elliptical Hough transfer, and peripapillary atrophy detection. By applying this elimination, edges with a higher probability of being non-disc structures, especially peripapillary atrophy, were excluded in order to

achieve higher segmentation accuracy.

in Mohan et al. (2018) proposed Fine-Net, which generates high-resolution optic disc segmentation of fundus images. Fine-Net is a CNN architecture with an emphasis on localization accuracy. The framework generalized well even in test images with high variability. Meanwhile, developed a deep retinal image understanding (DRIU) method in Maninis et al. (2016), which provides both retinal vessel and optic disc segmentation. Their approach utilizes multiple deep convolutional neural networks (CNNs). In particular, it uses a base network architecture, as well as two sets of specialized layers to detect retinal vessels and segment the optic disc.

2.2 Methodology

We now detail our novel scheme for training U-Net-style architectures Ronneberger et al. (2015). Our proposed approach consists of (1) restricting the training data to relevant examples and (2) training the network using dynamic cost functions. Below, we first provide a quick overview of the U-Net architecture, then discuss each training component, in turn.

2.2.1 *U-net architecture*

The U-net architecture is an *encoder-decoder* neural network with skip connections Ronneberger et al. (2015). In other words, it consists of a series of down-sampling layers followed by an equivalent number of up-sampling ones; the up-sampling layers match the size of the feature map from the down-sampling layer on the same level. This type of network can be applied to arbitrarily large images by scanning them across the full image. That is, one can partition the original image into *patches* that fit the network's input size. In this case, the

Table 2.1: **Dataset details**

Dataset	**Total#**	**Img dim.**	**OD dim.**	$\frac{OD}{Image}$	$(w \times h)$	$(w+p) \times (h+q)$
DRISTI Sivaswamy et al. (2014)	101	2049×1751	380×380	3.13%	836×836	1040×1040
DRIVE Staal et al. (2004)	40	565×584	80×85	1.79%	388×388	572×572
AV-WIDE Estrada et al. (2015b)	30	1237×809	75×80	0.40%	388×388	572×572
DRIONS-DB Carmona et al. (2008)	110	600×400	90×90	3.07%	388×388	572×572
CHASE-DB Fraz et al. (2012)	28	999×960	190×190	8.91%	388×388	572×572

network's input is generally slightly larger than its output in order to utilize the context around the region of interest. For example, if the output is of size $w \times h$, the input will be $(w+p) \times (h+q)$, where the extra $p \times q$ region is a $w \times h$ pixel extension beyond the $p \times q$ center along each dimension. If the region extends beyond the edge of the image, the standard practice is to mirror the pixels. As discussed below, we empirically determined the patch size for each dataset based on the average resolution of its images.

As in Khanal and Estrada (2020a), we use 3×3-pixel kernels except in the next-to-last layer, which has 1×1 kernels. The output layer is a *softmax* layer with two probabilities: one for a pixel being a vessel and another for it being part of the background. A pixel is labeled as either *vessel* or *background* based on the higher of the two probabilities. Here, though, we use half the number of filters in each U-net layer compared to Khanal and Estrada (2020a), for two reasons. First, unlike other segmentation problems (e.g., vessel segmentation), the optic disc only occupies a small portion of the image, thus requiring fewer features. Second, a smaller network can be trained faster and with less memory, allowing its use in less powerful hardware.

2.2.2 *Disc Centered Patch Augmentation*

As noted above, the optic disc (OD) comprises a small percentage of a standard retinal image. Specifically, Tbl. 2.1 shows that the average OD size can range from 9% down to less than one percent across existing datasets. This small size makes it harder to train a neural network since only a small region around the OD is actually useful for segmenting it. Training a network on patches across the entire image (e.g., Mohan et al. (2018)) forces the network to ignore a lot of noise in irrelevant patches in order to learn OD-specific features. To address this issue, we propose DCPA, a data selection method in which we select $r \times P$ random patches of size $w_i \times h_i$ from each image. Here, r is a ratio between zero and one (we used $p = 0.5$ for our experiments). For our final prediction, we picked the largest connected components from the network's output and filled any holes in this component via morphological operations. Importantly, we only use DCPA in the *training phase* since we empirically found that processing the entire image during testing was more robust. Below, we describe how we process images during both training and testing.

Patch selection in training

During training, we select $P \times r$ patches encompassing the entire optic disc; we randomly vary each patch's center location. We discard patches that have less than T (T=500 for our experiments) positive pixels in the ground truth because a network will learn little if there are very few positive pixels in a given patch. In addition to the OD, we empirically determined that the network also needs some patches belonging to the corners of the image to be more robust against borderline artifacts. In more detail, we first compute the *center-*

of-mass (c_x, c_y) of the OD in the ground truth. Then, we randomly shift this center of mass to $(c_x + a, c_y + b)$ and take a patch of size $w \times h$. Every time we sample this newly shifted center of mass, we ensure that the entire optic disc is within the original $w \times h$ patch (see Fig. 2.1). These random shifts increase variance in the training set as each time we sample a new patch, a new region of the image is fed to the network.

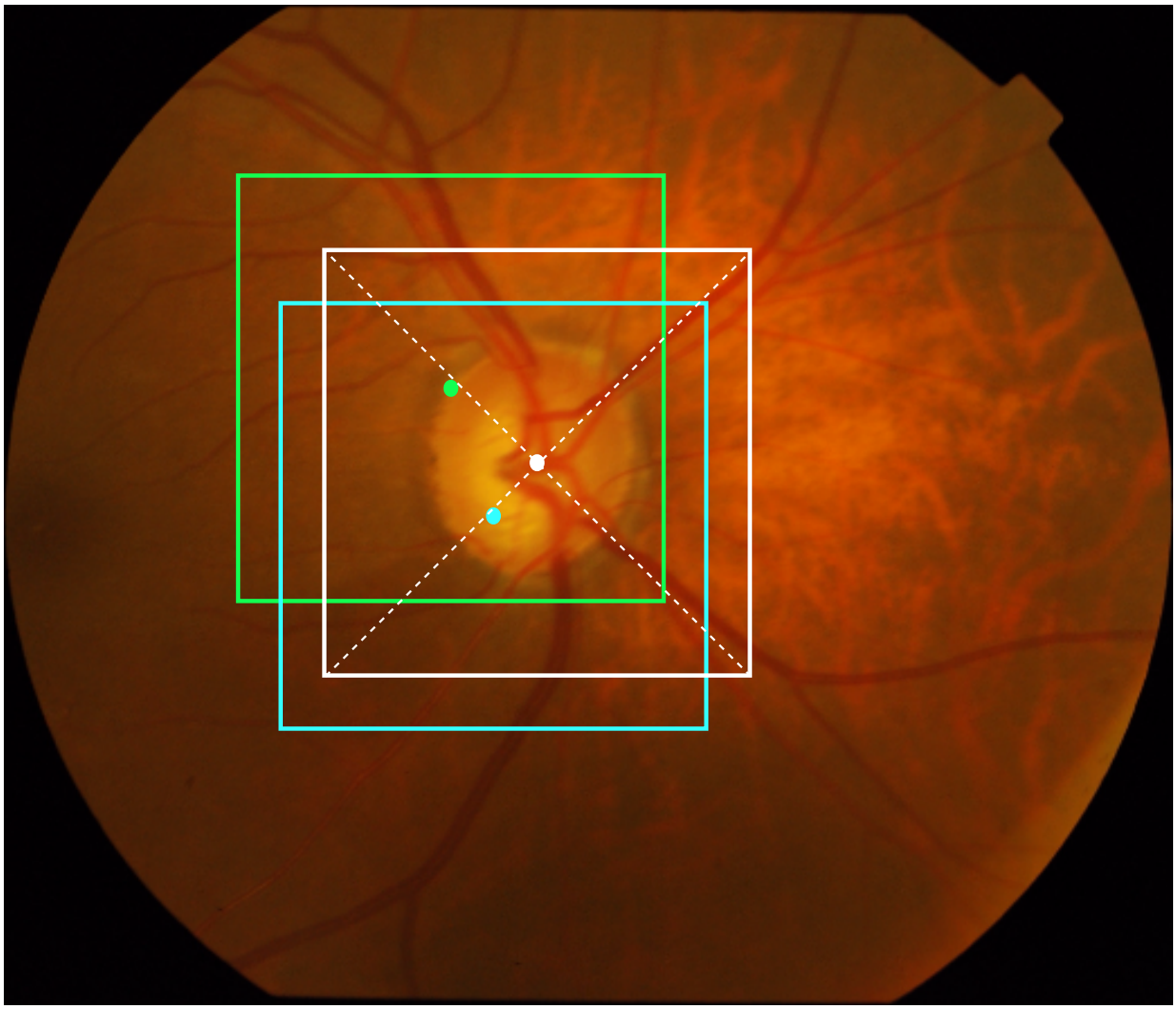

Figure 2.1: **Center of the mass shift to sample random patches for training phase:** We can see different patches picked at different sampling steps while training. The patch shown by the white boundary has the true center of mass of the manual segmentation. Other color-coded patches are sampled by shifting the center of mass by (a, b). The center of mass for each patch is color-coded.

Patch selection in evaluation

There are two ways one could feed a new image to a trained model: *(a).* Feed the entire image using a sliding window; *(b).* Use a preprocessing technique to identify the possible locations of the optic disc and just feed patches around those. Empirically, we found that feeding the entire image with a sliding window was significantly more robust and stable. As our experiments show, a trained network automatically ignores regions of the image that do not contain the OD.

2.2.3 *Stochastic cost functions*

Cross-entropy and dice loss are two widely used loss functions in supervised segmentation tasks. In particular, dice loss is widely used in medical image segmentation because, unlike accuracy, it is sensitive to high-class imbalances (i.e., where one class is much more likely than the other). However, as detailed in Khanal and Estrada (2020a), a network often performs poorly in ambiguous regions (i.e. regions with few positive pixels) when trained with these standard loss functions. Concretely, standard training leads to the network labeling only high-confidence pixels as positive. Thus, Khanal and Estrada (2020a) introduced a stochastic training scheme that forces the network to be more robust by randomly varying the misclassification penalty (i.e., the relative cost of a false positive vs. a false negative). In more detail, stochastic cross-entropy is defined as follows:

$$H = -\sum_{x} w_{rand(1,\alpha,s)} p(x) \log q(x), \tag{2.1}$$

where *rand* function draws the penalty parameter randomly from range 1 - α with a step-size of s to prevent exploding gradients. The stochastic version of dice loss uses the same additional parameters:

$$F_\beta = (1 + B_{rand}(1, \alpha, s)^2) \cdot \frac{precision \cdot recall}{B_{rand}(1, \alpha, s)^2 \cdot precision + recall}. \tag{2.2}$$

This stochastic approach, however, has only been applied to vessel segmentation. In this paper, we applied it to OD segmentation, and, as our experiments show, verified that it is also useful for this problem.

2.3 Experiments and Results

We empirically validated our DCPA and stochastic training schemes on five retinal datasets: DRISTI Sivaswamy et al. (2014), DRIVE Staal et al. (2004), DRIONS-DB Carmona et al. (2008), AV-WIDE Estrada et al. (2015b), and CHASE-DB Fraz et al. (2012) (see Tbl. 2.1). Below, we describe our equipment and experiments in more detail.

Hardware: The experiments of this paper were conducted on a Microsoft Azure VM on an Intel server with 16 cores, 512 GB RAM, and 4 Tesla V100 16 GB GPUs.

Datasets and ground truth preparation: As listed in Tbl. 2.1, we used five popular retinal datasets to assess our technique. The ground-truth OD segmentations for the DRISTI and DRIONS-DB datasets were available from the original authors since these two datasets were specifically designed for optic disc segmentation Carmona et al. (2008); Sivaswamy et al. (2014). The optic disc is centered in these images and covers a large portion of the image. In addition, we estimated the ground-truth OD segmentations for three additional retinal

datasets: DRIVE Staal et al. (2004), AV-WIDE Estrada et al. (2015b), and CHASE-DB Fraz et al. (2012). These datasets were originally created for vessel segmentation, so the OD ground truth was not available. Unlike the datasets created specifically for OD segmentation, the field of view in these images is not focused on the optic disc; it can appear on any corner of the image, making it a much harder problem.

Network training: We used 5-fold cross-validation for datasets that did not have separate training and test sets. For the ones with such separation, we used 30% of the images in the training set for validation. We used the ADAM optimizer Kingma and Ba (2014), a learning rate of 0.001, and a mini-batch size of 8 for all experiments. We used random vertical, and horizontal flips while training, and used the stochastic version of dice loss as explained in Sec. 3.2.2[1].

Network configurations: To better understand the impact of DCPA and stochastic weights, we tested four different network/training configurations on each dataset: (1) DCPA + stochastic weights; (2) DCPA only; (3)Stochastic weights only; (4) No DCPA or stochastic weights (standard training). Additionally, we used two network sizes for each set: $R1$, the full U-Net in the original paper Ronneberger et al. (2015), and $R2$, a network reduced by a factor of 2 on its width (i.e., number of channels reduced by 2 in each layer). This reduces the total parameters from 31 million to 7 million, making it possible to train without high-end configurations. Specifically, we were able to only train the $R2$ experiments on an Intel server with two 1080 Ti GPUs (11 GB of VRAM each) but not the $R1$. In our discussion below, we refer to fixed β vs. random β to indicate whether a configuration used stochastic weights.

Results: Figure 3.1 shows a sample result from each dataset.

[1]We also tested using cross-entropy, but the dice loss consistently yielded better results.

The precision, recall, F1 score, and overlap scores for the test images are listed in Tbl. 3.2; the latter is a common measure for OD segmentation corresponding to $\frac{TP}{TP+FN+FP}$. As noted above, the fixed-β + no DCPA configuration corresponds to a regularly trained U-net; the other configurations use DCPA, stochastic weights, or both. In addition, when available, we list prior state-of-the-art results.

Table 2.2: Experimental results: The method with boldface are different ablation studies. Column **R1** refers to the full-size model with 31M parameters, whereas **R2** refers to the reduced model(7M parameters). DCPA refers to the use of disc Centered Patch Augmentation, random beta refers to stochastic class weighted Dice loss function, and fixed beta refers to unweighted dice loss function.

Dataset	**Method**	R1 (31 million params.)				R2 (7 million params.)			
		Precision	**Recall**	**F_1**	**Overlap**	**Precision**	**Recall**	**F_1**	**Overlap**
DRISTI	Cheng *et al.* Cheng et al. (2011)	-	-	0.897	0.93	-	-	-	-
	FINE-Net Mohan et al. (2018)	-	-	**0.964**	**0.931**	-	-	-	-
	Fixed$_\beta$ + No DCPA	0.9291	0.9101	0.9195	0.851	0.9468	0.9269	0.9368	0.8811
	Fixed$_\beta$ + DCPA	**0.9565**	**0.9505**	**0.9535**	**0.9111**	**0.9531**	0.9471	**0.9501**	**0.9049**
	Random$_\beta$ + No DCPA	0.9184	0.9387	0.9284	0.8664	0.9502	0.9265	0.9382	0.8836
	Random$_\beta$ + DCPA	0.9553	0.9501	0.9527	0.9096	0.9452	**0.9504**	0.9478	0.9008
DRIVE	S. Roychowdhury Roychowdhury et al. (2016)	-	-	-	0.8067	-	-	-	-
	Bat Meta-heuristic Abdullah et al. (2018)	-	-	0.8810	0.8102	-	-	-	-
	Fixed$_\beta$ + No DCPA	0.8239	0.9306	0.874	0.7762	0.8669	0.9266	0.8958	0.8113
	Fixed$_\beta$ + DCPA	**0.8791**	0.9265	**0.9022**	**0.8218**	0.8767	0.9487	0.9113	0.8371
	Random$_\beta$ + No DCPA	0.8331	0.9515	0.8884	0.7992	0.87752	**0.9538**	0.9141	0.8418
	Random$_\beta$ + DCPA	0.8265	**0.9656**	0.8906	0.8028	**0.8962**	0.9362	**0.9158**	**0.8447**
DRIONS-DB	FINE-Net Mohan et al. (2018)	-	-	0.955	0.914	-	-	-	-
	Mannis *et al.* Maninis et al. (2016)	-	-	**0.971**	**0.944**	-	-	-	-
	Fixed$_\beta$ + No DCPA	0.9465	**0.9494**	0.9479	0.901	0.9637	0.9378	0.9506	0.9059
	Fixed$_\beta$ + DCPA	0.9627	0.9461	**0.9543**	**0.9126**	0.9274	**0.9467**	0.937	0.8815
	Random$_\beta$ + No DCPA	**0.9692**	0.9333	0.9509	0.9064	0.9441	0.9461	0.9451	0.8959
	Random$_\beta$ + DCPA	0.9631	0.9331	0.9478	0.9008	**0.9677**	0.9397	**0.9535**	**0.9111**
AV-WIDE	**Fixed$_\beta$ + No DCPA**	0.8399	0.6959	0.7611	0.6143	0.8576	**0.7444**	0.797	0.6625
	Fixed$_\beta$ + DCPA	0.9115	**0.7661**	**0.8325**	**0.7131**	**0.9287**	0.7315	**0.8184**	**0.6926**
	Random$_\beta$ + No DCPA	0.7479	0.7348	0.7413	0.5889	0.8581	0.6926	0.7665	0.621
	Random$_\beta$ + DCPA	**0.927**	0.7502	0.8293	0.7084	0.9207	0.723	0.81	0.6807
CHASE-DB	**Fixed$_\beta$ + No DCPA**	0.7388	**0.9341**	0.8251	0.7023	0.6928	0.9118	0.7874	0.6493
	Fixed$_\beta$ + DCPA	**0.7501**	0.9298	**0.8303**	**0.7098**	**0.7382**	0.9301	**0.8231**	**0.6994**
	Random$_\beta$ + No DCPA	0.7253	0.9126	0.8082	0.6782	0.7305	0.9286	0.8177	0.6916
	Random$_\beta$ + DCPA	0.7303	0.9305	0.8183	0.6925	0.7178	**0.9441**	0.8156	0.6886

For DRISTI, all eight network configurations achieved excellent results. The version with

the full R1 model, fixed-β with DCPA achieved state-of-the-art results across all metrics: a precision of 0.9565, recall of 0.9505, F1 score of 0.9535, and overlap of 0.9111; the last two scores are similar to the current state of the art Cheng et al. (2011); Mohan et al. (2018). Using DCPA improved the F1 score over 3% from 0.9195 to 0.9535 and the overlap score over 6% from 0.851 to 0.9111 on the full R1 model. We also observed over 2% improvement in F1 and overlap compared to the results reported in Mohan et al. (2018) (F1 score of 0.897 and overlap score of 0.93)[2]. Finally, in Cheng et al. (2011) report resizing their images, so we cannot directly compare our results to theirs.

For the DRIVE dataset, our DCPA-only version achieved over a 3% improvement in overlap and in F1 score over the state of the art. The full R1 model had actually slightly lower values (although still above the state of the art) which means that we don't need a full network R1 and can get a similar result with smaller network R2. In either case, we can see that DCPA plays a crucial role in improving results. Current state-of-the-art techniques Abdullah et al. (2018); Roychowdhury et al. (2016) reported an overlap of 0.8 and 0.81 respectively, whereas our technique obtained a score of 0.84. As noted earlier, we believe that the use of DCPA enabled the network to attend more to specific regions of an image where the optic disc is rather than try to encode the entire image. We can also see that using a random β adds more robustness to the performance of a neural network—a precision of 0.89 without random β vs 0.91 with random β, 0.86 vs 0.87 for F1 score, and 0.81 vs 0.84 for overlap.

For DRIONS-DB Carmona et al. (2008), we can see that using DCPA yielded slightly

[2]Based on the overlap formula, the overlap should always be lower than the F1 score, so we are not sure why to overlap is larger in this study.

better performance than FineNet Mohan et al. (2018), the current state of the art for this dataset. Even With the smaller R2 model, we obtained comparable results (0.9111 vs 0.9140 F1 score). In this case, using a random β in conjunction with DCPA yielded the best results.

For the AV-WIDE dataset Estrada et al. (2015b), the F1 score and overlap improved significantly with the use of DCPA (0.76 vs 0.83 F1, and 0.61 vs 0.71 IOU in the full model R1) (Table 3.2). This dataset was created for vessel segmentation and classification tasks. As such, it is more complex than the other datasets and has a much smaller disc-to-image ratio. Here, the OD only covers around 6% of the image patches as opposed to around 17% for the other datasets, making the problem harder. For the R2 model, we also observed an improvement of 2% on the F1 score when using the fixed-β and an improvement of about 5% on the F1 score when using random-β. Finally, the results on this dataset show that our technique is robust to different OD-to-image ratios, confirming that it generalizes better than other state-of-the-art techniques.

For CHASEDB Fraz et al. (2012), we can see that DCPA yielded more than 2% better recall (0.7388 vs 0.7501), about 1% better F1 score (0.8251 vs 0.8303), and a similar overlap (0.7023 vs 0.7098) compared to standard training (Table 3.2). In addition, the F1 score improved 5% by using DCPA on the reduced model R2.

Time results: Table 2.3 show the running time in seconds for the R1 and R2 network configurations on the various datasets. As expected, the running time for training the smaller models was considerably lower than the larger ones.

Table 2.3: Experimental run time in seconds for $R1$ and $R2$ in the same setting. The run-time reported is for one experiment for each dataset conducted in table 3.2 for full 351 epochs with batch size 8.

Model	DRISTI	DRIVE	DRIONS-DB	AV-WIDE	CHASE-DB
R1	14,200	1,530	3,760	21,800	21,600
R2	7,250	1,170	2,390	15,600	15,500

2.4 Discussion

We have shown that by using a novel training technique, we can consistently achieve state-of-the-art results using relatively small networks. This is very important because most researchers around the world do not have access to high-end training clusters as large labs and companies do. We can see in Tbl. 3.2 that DCPA improved or matched the results compared to a full U-Net network. We have shown such results in two of the widely used public dataset(DRISTI and DRIONS) and generated our own OD ground truth for three other datasets (DRIVE, AV-WIDE, and CHASEDB). As detailed in Sec. 5.3, DCPA restricts training to patches that contain the optic disc at different positions (to account for center-of-mass shifts). This ensures that the network is robust to OD position, as shown in our various results.

In particular, we obtained excellent results on datasets with the OD in the center (DRISTI, CHASE-DB, STARE, DRIONS), the right or left side (DRIVE), or with a very small disc-to-image ratio (AV-WIDE).

In addition, the random-beta stochasticity described in paper Khanal and Estrada (2020a) enables the network to be more robust against ambiguous pixels around the border of the

OD, which consistently improved recall as shown in Tbl. 3.2. For example, we can see that stochasticity, along with DCPA, yielded higher recall on DRIVE. However, for other datasets, such as DRIONS-DB, and CHASE-DB, the use of DCPA was sufficient without stochasticity. As mentioned earlier, in vessel segmentation Khanal and Estrada (2020a) the vessels are well scattered across the image. Thus, the network needs to learn a wide variety of shapes, which made stochastic penalties very useful. This robustness was not as crucial in the case of the optic disc, since it is usually a blob-like shape in a fixed location. As such, the use of DCPA was often sufficient to achieve state-of-the-art results.

2.5 Conclusions and Future work

In our work, we developed a novel approach for segmenting the optic disc in retinal fundus images. Our extensive comparison with the state of the art, across five datasets, showed that our technique consistently achieves excellent precision, recall, F1, and overlap scores. In addition, our ground-truth segmentations for three of the datasets (CHASE-DB, DRIVE, AV-WIDE) can be used as a benchmark for future researchers. Overall, our results suggest that it is important to have patches that contain the entire optic disc, with variations in positions, in order for the network robustly segment the OD across different imaging devices, zoom factors, image quality, and device resolution. For future work, we aim to use this work for diagnosing glaucoma and other retinal diseases that affect the optic disc.

Data Availability Statement

The source code and datasets generated for this study can be found in the following GitHub repository: `https://github.com/saeidmotevali/fundusdisk`.

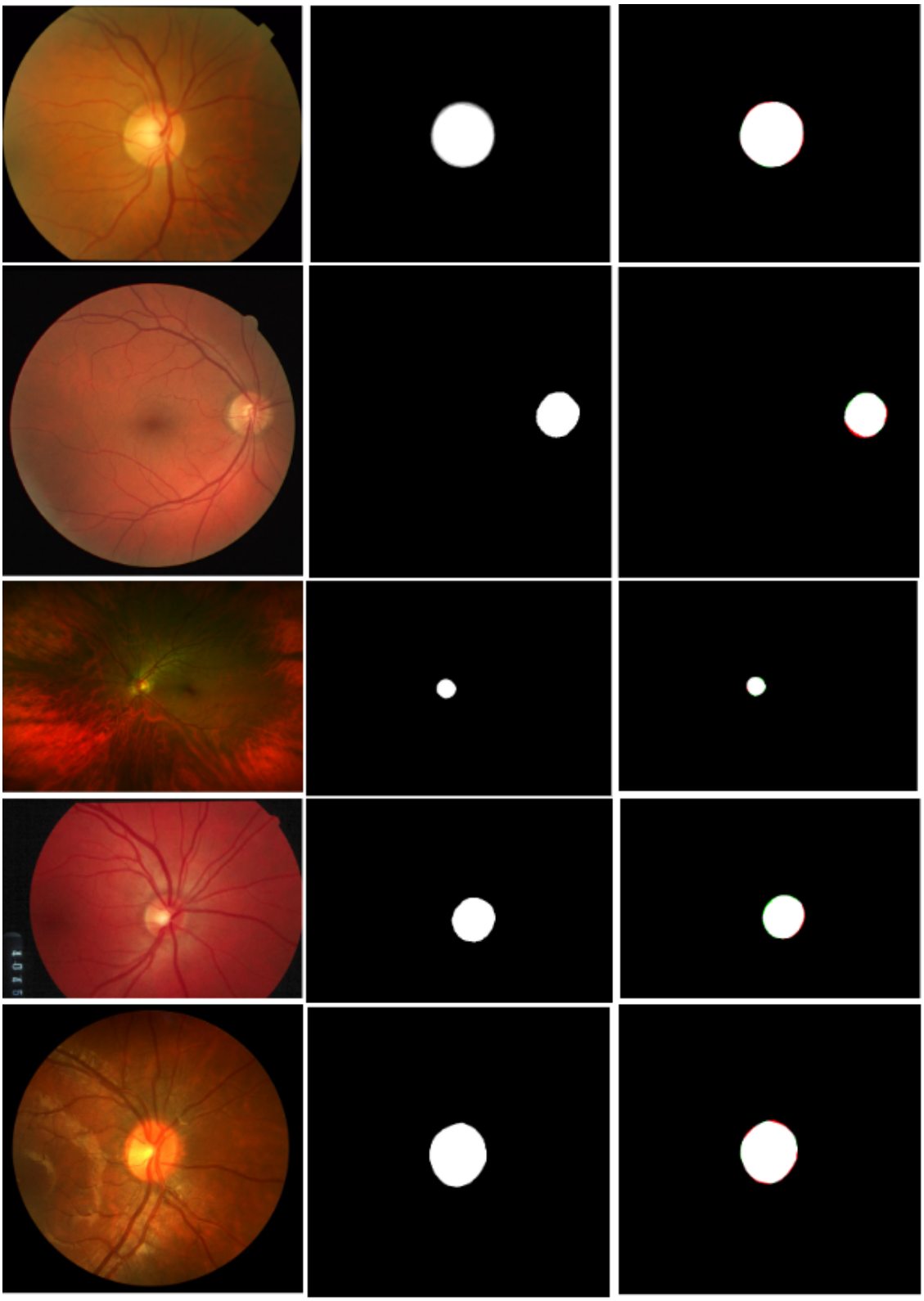

Figure 2.2: **Segmentation results on different datasets(top to bottom) with image, segmentation ground truth, and our segmentation result (left to right):** DRISTISivaswamy et al. (2014), DRIVEStaal et al. (2004), AV-WIDEEstrada et al. (2015b), DRIONSCarmona et al. (2008), CHASE-DBFraz et al. (2012). The white pixels in segmentation result represent True Positive(TP)/Optic Disc, green pixels represent False Positive(FP), red pixels represent False Negative(FN), and black pixels represent True Negative(TN)/background

3 | Fundus Optic Cup Segmentation

In the previous chapter 2, we presented our research on optic disc segmentation. In this chapter, we present our optic cup segmentation. By having these two segmentations, we can get a cup-to-disc ratio, which is an important factor in glaucoma detection Mangipudi et al. (2021). These two chapters 2, 3 are part of an attempt at non-vascular fundus image analysis.

3.1 Introduction

The optic disc and optic cup segmentation are independent but indispensable in medical image analysis Harizman et al. (2006). This segmentation is currently performed by professionals. This segmentation is a very time-consuming process. The computer can perform faster segmentation for fundus images Joshua et al. (2019). In general, optic cup segmentation is harder than optic disc segmentation. First of all, in the cup segmentation, we are looking for a smaller object. Plus, the boundaries of the optic cup are more ambiguous, which makes it harder to perform the segmentation. The optic cup segmentation was only available for DRISTI Sivaswamy et al. (2014) dataset, and that is why we only did the optic cup segmentation for the DRISTI dataset. We created the ground truth for the optic disk but creating such a ground truth for the optic cup was beyond our expertise. The optic

Table 3.1: **Dataset details**

Dataset	**Total#**	**Img dim.**	**OC dim.**	$\frac{OC}{Image}$	$(w \times h)$	$(w+p) \times (h+q)$
DRISTI	101	2049×1751	315×315	2.77%	836×836	1040×1040

cup is smaller in area, as shown in 3.1. The optic cup area is about only two percent of the area of the fundus image, which makes it very challenging to do segmentation. Also, it is harder to define boundaries for the optic cup. That is why we were not able to create ground truth for datasets where the optic cup ground truth was not available. After doing the segmentation for the optic disc and optic cup, we can get a cup-to-disc ratio which is an important factor in predicting diseases such as glaucoma Mangipudi et al. (2021).

3.2 Methodology

We use the very well-known U-net architecture for optic cup segmentation Ronneberger et al. (2015). Just like optic disc segmentation, our proposed approach consists of (1) restricting the training data to relevant examples and (2) training the network using dynamic cost functions. We used U-net architecture for cup segmentation. The U-net architecture is an *encoder-decoder* neural network with skip connections Ronneberger et al. (2015). In disc segmentation, we used half a number of filters in each U-net layer (R2), and we have shown that the model can produce almost identical results as standard U-net (R1). In optic cup segmentation, defining the boundary is very complex, and the R2 model does not produce reasonable results.

3.2.1 *Cup Centered Patch Augmentation*

As noted before, the optic cup contains a very small percentage of a retinal image 3.1. Detecting and segmenting this small area is a very hard task for a model. Training a network on patches across the entire image (e.g., Mohan et al. (2018)) forces the network to ignore a lot of noise in irrelevant patches in order to learn OC-specific features. To address this issue, we propose DCPA, a data selection method in which we select $r \times P$ random patches of size $w_i \times h_i$ from each image. Here, r is a ratio between zero and one (we used $p = 0.5$ for our experiments).

Patch selection in training

During training, we select $P \times r$ patches encompassing the entire optic cup; we randomly vary each patch's center location. We discard patches that have less than T (T=500 for our experiments) positive pixels in the ground truth because a network will learn little if there are very few positive pixels in a given patch.

Patch selection in evaluation

Just like optic disk detection, we used a preprocessing technique to identify the possible locations of the optic cup. Then we just feed patches around those. Empirically, we found that feeding the entire image with a sliding window was significantly more robust and stable. As our experiments show, a trained network automatically ignores regions of the image that do not contain the OC.

3.2.2 *Stochastic cost functions*

We used dice loss which is widely used in medical image segmentation because, unlike accuracy, it is sensitive to high-class imbalances (i.e., where one class is much more likely than the other).

3.3 Results and future work

We got an acceptable result for our DRISTI optic cup segmentation. We hope that in the future, more ground truth will be available for optic cup segmentation.

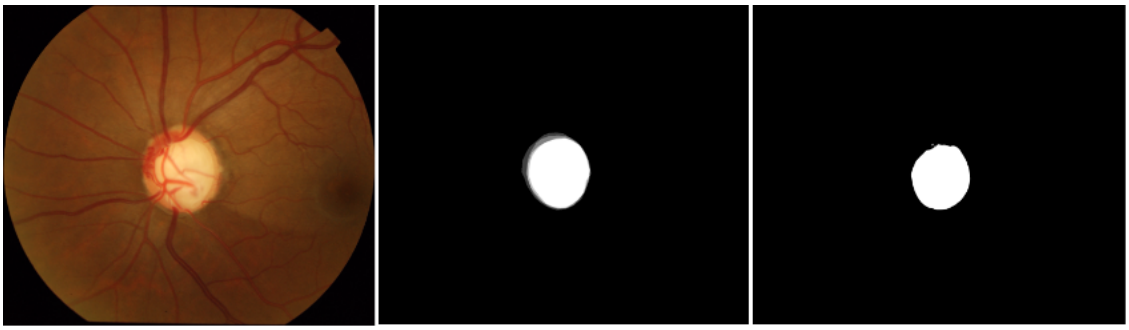

Figure 3.1: **Segmentation results on DRSITI datasets, with image, segmentation ground truth, and our segmentation result (left to right):** DRISTISivaswamy et al. (2014). The white pixels in the segmentation result represent True Positive(TP)/Optic Disc, and the black pixels represent True Negative(TN)/background

Table 3.2: Experimental results: The method with boldface are different ablation studies. Column **R1** refers to the full-size model with 31M parameters. DCPA refers to the use of Disk Centered Patch Augmentation, random beta refers to stochastic class weighted Dice loss function, and fixed beta refers to unweighted dice loss function.

Dataset	**Method**	R1 (31 million params.)			
		Precision	**Recall**	$\mathbf{F}_1$	**Overlap**
DRISTI	**Fixed$_\beta$ + No DCPA**	0.8310	**0.9400**	0.8821	0.7891
	Fixed$_\beta$ + DCPA	0.86256	0.9121	0.8867	0.7964
	Random$_\beta$ + No DCPA	0.79821	0.9351	0.8613	0.7563
	Random$_\beta$ + DCPA	**0.8796**	0.9219	**0.9003**	**0.8186**

4 | Attention Score in Brain MRI

In the previous chapters, we worked with fundus retinal images, which are 2D images. In this chapter, we work with brain-MRI 3D images, which is a good extension and a transformation from 2D to 3D. Unlike the fundus retinal images, which are colored images, the brain-MRI images are in gray format.

4.1 Introduction

Attention score is an important number in children's psychology evaluation. In the current setup, there is a questionnaire, and the attention score is determined based on the children's answers to that questionnaire Shen et al. (2020). In this research, we tried to predict the attention score solely by using 3-D brain MRI images without using the questionnaire. Predicting the attention score solely based on 3-D MRI images is very complex. The data is very skewed, and it makes it very hard to make the prediction. We also made this task simpler by converting the problem to a classification problem in which one class represent normal kids and the other class represents kids with high attention score. We also created a model to do the gender and age prediction. We also used the weights from age and gender prediction and used them for transfer learning for attention score prediction and classification. We use the ABCD dataset for all attention score predictions. The visualization of 3-D MRI images

is shown in Figure 4.1

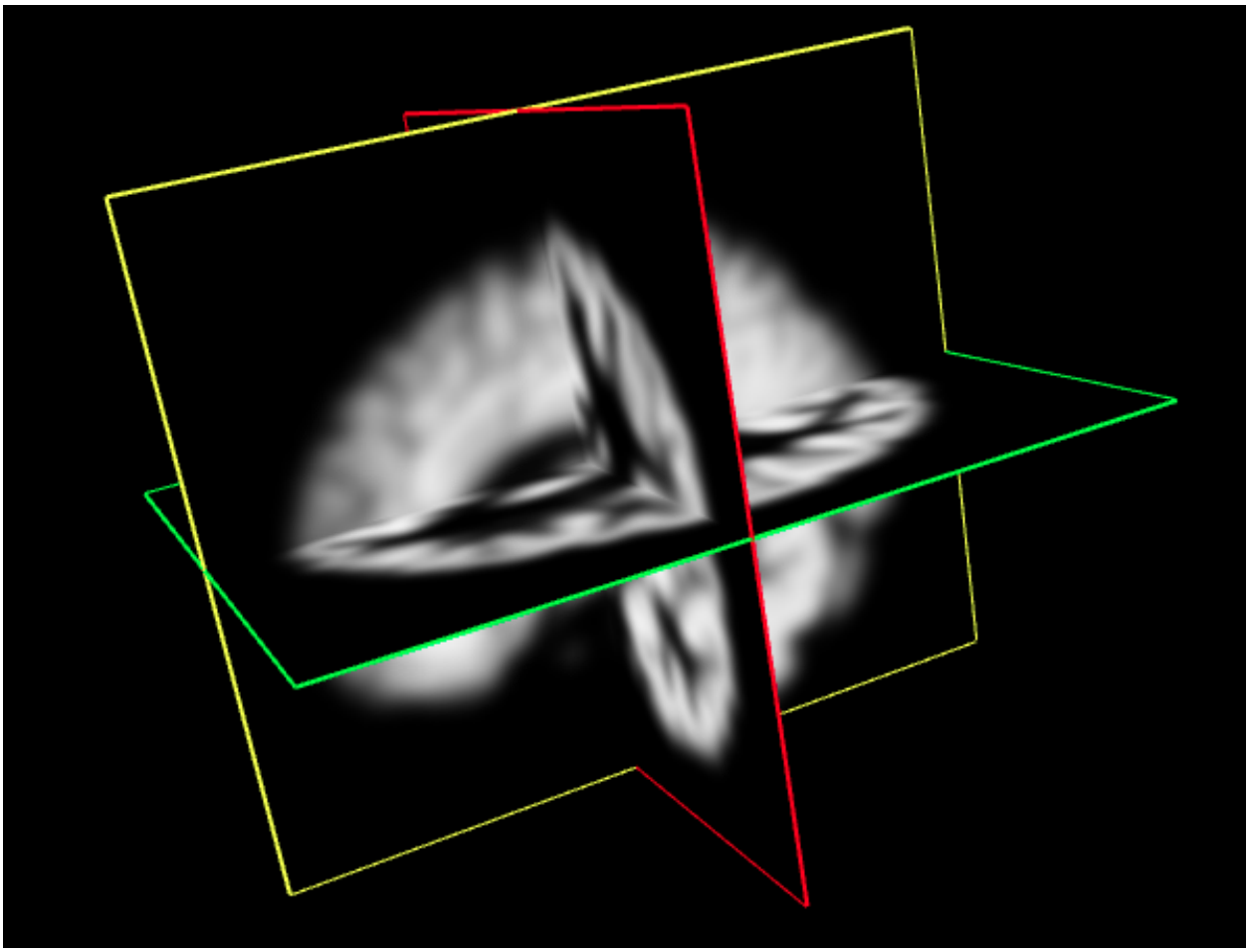

Figure 4.1: **Brain MRI visualization**

4.2 ABCD data

ABCD Data Repository holds the Adolescent Brain Cognitive Development data. In the ABCD study, start monitoring children at the age of 9-10, then follow up with the participant for ten years. The data were collected at 21 research facilities across the nation, and around 12,000 youth participated in the program. It does measure social, emotional, cognitive development, and mental health. It also detects substance use and attitudes, gender identity, sexual health, biospecimens, a variety of physical health, and environmental

factor. Moreover, it detects brain structural development, task functional, and resting state functional imaging. Several external databases also conducted research connected to the ABCD database, including local conditions for the environment, poverty, pollution, school, and policy. The NIH partners, including the National Institute on Drug Abuse, the National Institute on Alcohol Abuse and Alcoholism, the National Cancer Institute, the National Institute of Mental Health, The National Institute of Neurological, and many more governmental and research institutes, participated and supported this database. nih (1999) The ABCD dataset has 11,864 samples of kids aged between 107 to 133 months old. An attention score is a number between 50 to 100, which is determined based on the child's answers to the questionnaire provided by professionals. A normal child should score fifty or close to fifty, and a higher number indicates that the child has an attention problem.

4.3 Data Preparation

After the initial review of the dataset, we realized that some of the data samples in ABCD MRI images do not have sufficient values. Some of the images are very dark or very bright, which can not be used in machine learning data modeling. Those samples of 3-D MRI images were affected, and we took them out of the dataset before we started the experiments. That brings down the number of samples to 11,714 from the original 11,864 samples. When we trained the model, we used the entire 3-D MRI images. We did not modify or use part of the MRI images. Since the data is in the 3-D format, we used 3-D convolution to process the data. One of the main problems we faced was the fact that the data was very skewed. We have 4,694 samples with an attention score value of 50 and 1,539 samples with an attention score value of 51. We only have 284 samples whose attention score value is above 70. This

makes a dataset very skewed and biased toward an attention score close to 50.

4.4 Models

After we made sure that the data was in the right format, it was important to use the right machine-learning model for our prediction. A very shallow model can easily get overfit, and a very large model takes a long time to get trained if it ever gets trained. In this experiment, we tried different models and reported our observations on each model. Visual Geometry Group (VGG) is a standard deep Convolutional Nural Network. The VGG-16 model is one of the famous models in machine learning. VGG, also known as VGGNet, is a classical convolutional neural network (CNN) architecture. VGG increased the depth of such CNNs in order to boost the performance of the model. The word deep refers to the number of convolutional layers which we have in the model. In VGG-16, we have 16 convolutional layers, and in VGG-19, we have 19 convolutional layers. Researchers use VGG architecture for ground-breaking object recognition. Deep learning growing rapidly in the last few years and has been widely used in so many different areas, such as image recognition, computer vision, speech recognition, machine translation, medical imaging, robotics, and many more. In most cases, deep learning outperformed traditional machine models.

VGG16 was proposed by A. Zisserman and K. Simonyan from the oxford university. The VGG16, which is a convolutional neural network model, was titled "Very Deep Convolutional Network for Large-Scale Image Recognition." It was one of the most popular models at ILSVRC-2014. This model achieved 92.7% top-5 test accuracy in ImageNet. ImageNet contains more than 14 million images in 1000 classes. In AlexNet, a large kernel-sized filter was used, and these large kernel-sized filters were replaced with several 3×3 kernel-sized

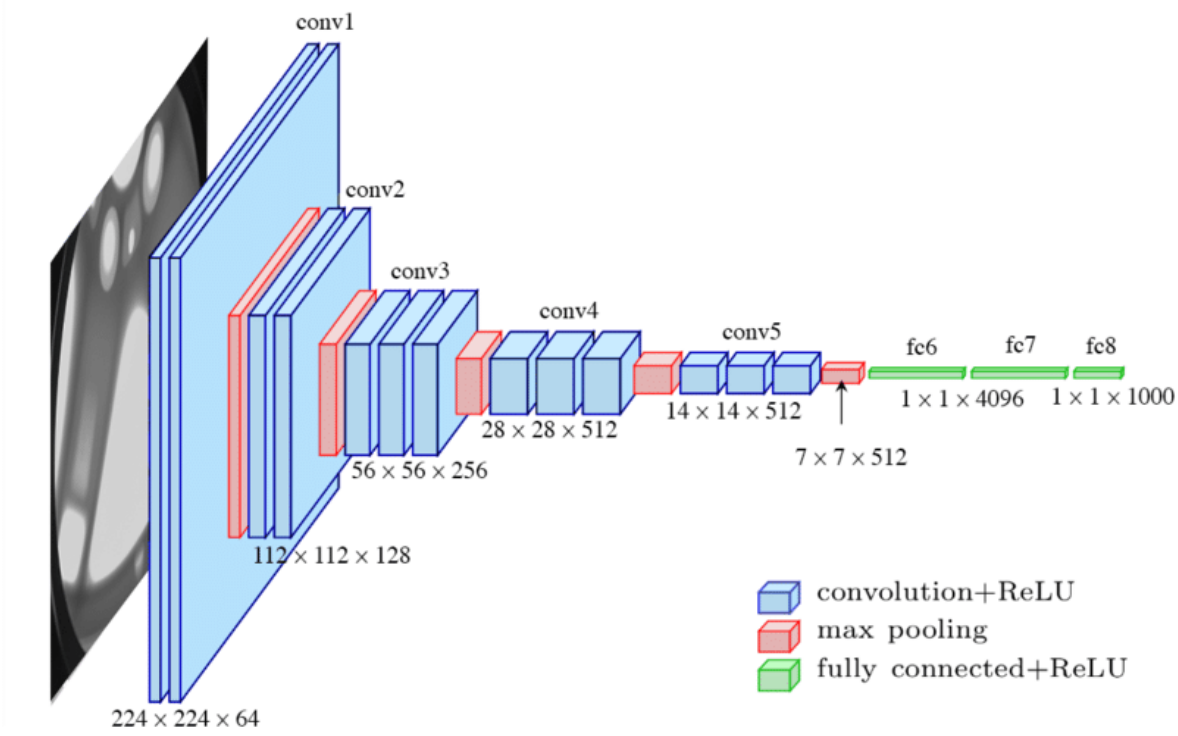

Figure 4.2: **VGG-16 detailed architecture Ferguson et al. (2017)**

filters. It made significant improvements over AlexNet. The Nvidia Titan Clack GPUs were used to train the VGG-16 model to classify 1000 object categories. The input image size was 224-by-224 for this training.

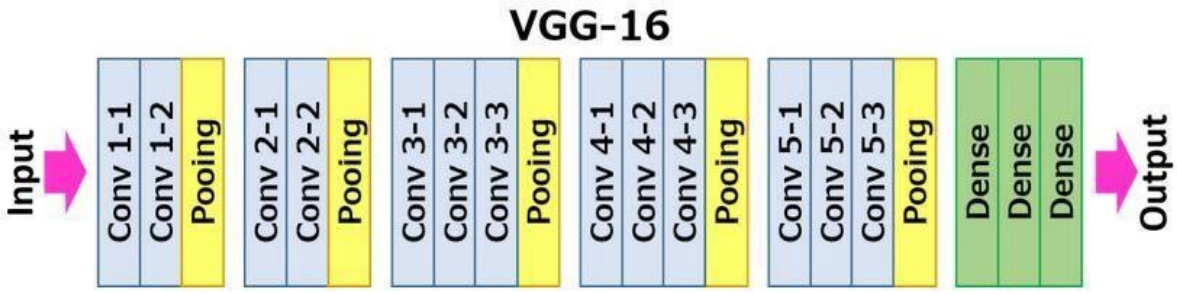

Figure 4.3: **VGG-16 convolution layers**, `https://neurohive.io/en/popular-networks/vgg16/`

In VGG, we have many small convolutional filters. In VGG-16, we have thirteen convo-

lutional layers followed by three fully connected layers. In the ImageNet competition, their input image size for VGGNet was 224×224, and the size was consistent for each image. As far as the convolutional layers, VGG convolutional layers leverage the minimal receptive field. The size of the convolutional layer is 3×3 which is the smallest possible size that still can capture information in all directions. There are also 1×1 convolution filters followed by the ReLu unit, which reduces the training time compared to AlexNet. Rectified linear unit activation (ReLU) was used, which will output the input if positive and makes it zero if negative. ReLU is used in all hidden layers of the VGG network. VGG usually does not use Local Response Normalization (LRN) since it increases the training time and memory consumption, especially since it does not increase the overall accuracy. In the Fully-connected layer, VGG has three fully connected layers. The last layer has 1000 channels which match the number of classes that the VGGNet model needs to predict. The prior layers have 4096 channels each. In VGG16, the number 16 refers to the number of the convolutional layer in the model. This is a pretty heavy model with around 138 million parameters. This is a huge model in terms of the number of parameters in the modern standard. At the same time, VGG-16 is a relatively simple model in terms of the elements used to create the model. That makes the model very interesting. This model has a few convolutional layers followed by a pooling layer. Also, we have the option to use 64, 128, 256, or 256 filters for the last layer. VGG-16 model has about 138 million parameters, and it takes a long time to train a model with that many parameters because of the depth of the model and the number of fully connected layers in the model. The VGGNet uses backpropagation so that if the number of layers is very high, the changes to the initial layers are very small. If the number of layers increases in VGGNet, the training time will increase significantly. The alternative to this is

to use ResNet. ResNet is less complex than VGGNet since it has fewer filters. The ResNet does not allow vanishing gradient problems which enables the ResNet to come in a deeper format such as ResNet 50, ResNet101, and ResNet152.

We use the VGG-16 model, which has five convolutions layers followed by three linear layers at the end. When we used the VGG-16 model for transfer learning, we only applied one linear layer at the end of each model. We also used ResNet to check and see if that improved the model's performance. Our observation showed that the ResNet is too big for this kind of task, and we did not observe any improvement in the model performance. VGG-16 was the default model for our experiment, and when we used other models, we specifically mentioned it.

4.5 Age Prediction

In our first step, we wanted to test our system for easier tasks and see if we could get similar to the state-of-the-art result. That is why we tested our pipeline for age and gender prediction. We used the same input sample size as we used for attention score prediction. This means we use 11,714 samples as input data for our system. The structure of the model used for age prediction was also the same as the attention score prediction model. We used the same VGG-16 which we used for attention number prediction. Regarding the method to calculate the error, we used MSE error for age prediction, and since the gender prediction was a classification problem, we used Cross Entropy to calculate the error. For age prediction, our MSE error was 56.0, which is a reasonable number. But we wanted to check and see how adding weight to each level of our architecture can lower the MSE error. We made the model 10 times bigger on each layer and trained a model with more parameters and as a

result, our MSE error went down to 51.7.

Age Prediction	
ABCD Dataset	**MSE**
All Data	**56.00**
All Data + 10 W	**51.71**

4.6 Gender prediction

In this step, we tested our pipeline and wanted to evaluate how our model works for gender prediction. For gender prediction, we also used the same input and sample size as we used for attention score predictions. Overall we used the entire sample size of 11,714, which was available on the ABCD dataset. We only took out the samples which do not have useful information in them, but the rest of the samples were used in our prediction. As far as the model for gender prediction, we use what we used for attention score predictions. The VGG-16 is what we used for gender prediction. We also used ResNet for gender prediction, but we did not observe any significant differences between the two models. For gender prediction, we ran the model for different learning rates such as 0.01, 0.05, 0.001, 0.005, 0.0001, and 0.0005, and we got an accuracy of 0.94, F1 score of 0.94, Precision of 0.93 and recall of 0.94 when the learning rate was 0.0005. For this experiment, we used five folds cross-validation when we trained our model. We also wanted to compare the error when we used a much larger model with more parameters. That is why we made each layer of our VGG-16 model

10 times bigger, which will increase the number of parameters that we train in our model.

ABCD Dataset(Gender Classification)	Accuracy	F1	Precision	Recall
Gender Classification, Learning rate 0.01	**92.48**	**92.76**	**92.1**	**93.42**
Gender Classification, Learning rate 0.05	**89.42**	**90.08**	**87.33**	**93.01**
Gender Classification, Learning rate 0.001	**92.63**	**92.21**	**93.49**	**90.96**
Gender Classification, Learning rate 0.005	**92.78**	**93.11**	**93.68**	**92.55**
Gender Classification, Learning rate 0.0001	**91.60**	**92.10**	**89.42**	**94.95**
Gender Classification, Learning rate 0.0005	**93.55**	**93.66**	**92.59**	**94.76**
Gender Classification, Learning rate 0.005, 10 W	**93.82**	**93.77**	**93.20**	**94.34**

In our experiment, we used five folds cross-validation which is a general practice in the field. But we also wanted to test and see if ten folds cross-validation may have a significant impact on the model and if it will increase the accuracy, F1 score, precision, or recall. As you can see in the figure, it did not have a significant effect on the overall performance of the model. That is why the five folds cross-validation is what we used for our experiment.

ABCD Dataset	Accuracy	F1	Precision	Recall
Gender Prediction, Learning rate 0.001, 10 Folds	**90.58**	**91.36**	**92.46**	**90.28**

ResNet

We also tested our system with a ResNet model. We know that ResNet has more parameters, and it is deeper than our VGG-16 model. We did not observe a significant increase in the

performance of the model. That is why we did not use ResNet for our experiment. We have an F1 score of 91.64 for ResNet with a learning rate of 0.005 when using the five-fold cross-validation, and we got an F1 score of 93.11 with the same setup with the VGG-16 model. That is why did not continue our experiment with the ResNet model.

ABCD Dataset, ResNet	**Accuracy**	**F1**	**Precision**	**Recall**
Gender Prediction, Learning rate 0.005, 5 Folds	**91.84**	**91.64**	**89.7**	**93.67**

4.7 Attention score Classification

We realized that is hard for the model to learn to predict the attention score value. That is why we changed the task to classification rather than the usual attention score value prediction. The classification is typically easier for the network to learn. We used the same pipeline and VGG-16 model for our training. In our first attempt, we put the samples with an attention score of 50 in one class, and the other class contains the samples with an attention score of more than 69. An accuracy of 0.40, an F1 score of 0.07, a precision of 0.04, and a recall of 0.6 was achieved, which obviously shows the network did not learn to perform the required classification. In another attempt, we created a class with an attention score below and another class with an attention value of more than 60. We got an accuracy of 0.75, an F1 score of 0.14, a precision of 0.11, and a recall of 0.18. In another attempt, we created two classes. Sample with attention scores of 50 and 51 in one class and samples with attention scores of more than 51 in another class. We got an accuracy of 0.48, F1 of 0.65, precision of 0.48, and recall of 0.99.

ABCD Dataset (Attention score classification)	Accuracy	F1	Precision	Recall
Attention score classification 50, 51 and Above 51	**48.22**	**64.94**	**48.18**	**99.58**
Attention score classification 50 and Above 69	**40.46**	**7.17**	**3.81**	**60.00**
Attention score classification 50 and Above 69, 50 deleted, data balanced	**48.53**	**62.17**	**48.68**	**86.00**

4.8 Attention Score Prediction

After the initial cleaning of the data, we tried to predict the attention score solely by using 3-D brain MRI images. After the initial observation of the data and tossing off some of the sample data, which does not contain useful information, the rest of the data (11,714 samples) was used for this prediction. We also used the VGG-16 model for this prediction. The mean squared error (MSE) was 41.65, which is a high error given the fact that the attention score ranged only from 50 to 100. We also trained the model when we used only the data with an attention score value less than 60, and the MSE was 6.78, which is a high error. We also trained the model when an attention score was less than 60. We also trained the same model, but we took out the data with an attention value score equal to 50. Our MSE was 5.5 in this experiment. Again this is a kind of high error for the ground truth variant is only 9.

Attention score Prediction	
ABCD Dataset	**MSE**
All Data	**41.65**
Equal Bins	**75.90**
Attention Score Less than 60	**6.78**
Attention Score Less than 60 except 50	**5.5**

4.9 Transfer learning

After our attempt at attention score value prediction and classification, we realize it is difficult for the models to learn such a task. At the same time, we have a good result for age and gender prediction. We tried to use the feature we learned and use those features to see if they have any effect on attention score prediction. Since we had a state-of-the-art result in gender prediction, we used that model and extracted the features from that model, and used it for attention score prediction. In our first attempt, we tried to predict the attention score by using the features we extracted from gender prediction. For this experiment, we trained the model for gender prediction and use these weights as a preweight for attention score prediction. Then we retrain the model using the weight from gender prediction as a preweight for our models. We used the structure of the VGG-16 for this experiment. The learning rate (LR) was 0.005 for the entire transfer learning experiment. We also adopted

more parameters in each layer. Our weight (W) is equal to 10, which means we multiplied each layer of the model by 10. That makes the model a lot bigger in terms of the number of parameters that we need to train. The batch size is 56 for this experiment. The data is skewed, and it is not balanced. The majority of data is around attention number 50, which represents the normal kids. In order to have a better prediction, we divided the data into 10 equal bins based on their attention score number and randomly selected 50 and 100 samples for each bin. We run this experiment for both 5 and 10-fold cross-validations. The MSE error was 40.24 when we ran the model for all data. Even though this result looks fine, in fact, it does not predict the attention score with higher values correctly, which means the model pretty much predicts a number close to 50, which is the attention score average. Because there is a lot of samples with normal attention score, which is 50 or 51. When we used equal bins(10 equal bins from 50 to 100) and we selected only 50 samples per bin, the attention score MSE error was higher, but we had a better prediction for samples with high attention scores. The error went down when we used 10 folds cross-validation instead of 5 folds cross-validation. But when we used the equal bins and we selected 100 samples for each bin, the result of 5 and 10 folds cross-validation was almost the same, which indicates that we have enough data to train the model, but that was not the case for bins with 50 samples.

ABCD Dataset(Transfer Learning, MSE Error)	MSE
Att score predict, LR 0.005, 10 W, After gender, 10 folds	**40.24**
Att score predict, LR 0.005, 10 W, After gender, equal bins 50 samples, 5 folds	**121.20**
Att score predict, LR 0.005, 10 W, After gender, equal bins 50 samples, 10 folds	**87.83**
Att score predict, LR 0.005, 10 W, After gender, equal bins 100 samples, 5 folds	**95.65**
Att score predict, LR 0.005, 10 W, After gender, equal bins 100 samples, 10 folds	**96.30**

Attention score prediction is a complex task. We need to predict just one number from 50 to 100 based on the Brian-MRI images. To simplify this task, we classified the attention score into two classes. One class contains an attention score value of 50 to represent normal kids, and the other class contains an attention score above 69, representing kids with attention problems. We already reported the result for such a classification, but in this section, we used transfer learning to do the same task. For this experiment, we classify the kids into two classes. One with attention problem, and the other one represent normal kids. We trained the model to predict the kid's gender and age prediction. Then used this model as a preweight to classify the data. We used VGG-16, but we modified the original VGG-16, and instead of three linear layers after the convolutional layers, we just have one linear layer after the convolutional layers. We freeze all layers of the model and let the model retrain, and just update the weight for the last few layers of the models. In the first attempt, we freeze all layers and let the model only modify the linear layer (-1 L). Then in the second attempt, we again freeze all of the layers in the model and let the model retrain and modify the weight of the two last linear layers, which includes the linear layer (-1 L) and the last convolutional layer (-1 Con). In this case, we show it as (-1 L -1 Con), which means we modified the last

linear and convolutional layers. If we retrain and modify the weights for the last linear layer followed by the last two convolutional layers of the model, we show it as (-1 L + -2 Con) to represent that. The learning rate (LR) was 0.005 for this experiment. The batch sizes were 16, 32, 56, and 64. We also applied weight and multiplied the number of parameters in each model layer by 10. This makes the model a lot bigger in terms of the number of parameters, and it takes longer to retrain the model. We used the same architecture throughout the experiment. When we tried to classify the data into two classes, the data was not as skewed as when we wanted to predict the attention score, but still, the data was very imbalanced. In order to balance (represented with B in the table) the data, we randomly selected the same number of samples for each class. As shown in the table, the model has a very good recall score, and the F1 score is also resealable in some cases, but since the accuracy is very low, that indicates the model does not have a good performance.

ABCD Dataset classification (50 and above 69)	**Accuracy**	**F1**	**Precision**	**Recall**
Att score classify, LR 0.005, 10 W, 64 b, 50 and less 69, After Gender	**40.46**	**7.17**	**38.12**	**60.00**
Att score classify, LR 0.005, 10 W, 64 b, 50 and less 69, After Gender, B	**48.53**	**63.17**	**48.68**	**86.00**
Att score classify, LR 0.005, 10 W, 56 b, 50 and less 69, After Gender, -1 L, B	**56.13**	**70.69**	**55.78**	**96.47**
Att score classify, LR 0.005, 10 W, 56 b, 50 and less 69, After Gender, -1 L -1 Con, B	**54.84**	**70.83**	**54.84**	**1**
Att score classify, LR 0.005, 10 W, 64 b, 50 and less 69, After Age, -1 L, B	**54.19**	**68.72**	**54.93**	**91.77**
Att score classify, LR 0.005, 10 W, 32 b, 50 and less 69, After Age, -1 L, B	**52.9**	**68.26**	**54.14**	**92.35**
Att score classify, LR 0.005, 10 W, 16 b, 50 and less 69, After Age, -1 L, B	**53.87**	**67.57**	**54.98**	**87.65**
Att score classify, LR 0.005, 10 W, 64 b, 50 and less 69, After Age, -1 L -1 Con, B	**53.55**	**69.49**	**54.31**	**96.47**
Att score classify, LR 0.005, 10 W, 64 b, 50 and less 69, After Age, -1 L -2 Con, B	**52.9**	**64.04**	**55.09**	**76.47**

As we mentioned earlier, one of the most challenging parts of this experiment is dealing with skewed data. In order to solve this problem and have more balanced classes, we changed the class category to check the model's performance. For this experiment, we have data with an attention score value of 50 in one class and data with an attention score of greater than

63 in another class. As shown in the table, we have a high recall, but since the accuracy is very low, it means that the model does not have a good performance.

ABCD Dataset (50 and above 63)	Accuracy	F1	Precision	Recall
Att score classify, LR 0.005, 10 W, 32 b, 50 and less 63, After Gender, -1 L, B	49.50	65.26	48.93	97.94
Att score classify, LR 0.005, 10 W, 64 b, 50 and less 63, After Gender, -1 L, B	49.32	65.39	48.85	98.88
Att score classify, LR 0.005, 10 W, 64 b, 50 and less 63, After Gender, -1 L -1 Con	68.07	20.91	19.69	22.3
Att score classify, LR 0.005, 10 W, 64 b, 50 and less 63, After Gender, -1 L -1 Con, B	48.60	65.24	48.50	99.66
Att score classify, LR 0.005, 10 W, 64 b, 50 and less 63, After Gender, -1 L -2 Con, B	48.87	64.80	48.60	97.20

4.10 Conclusion

We tried many different experiments, and we did not find any significant correlation between the brain-MRI images and the attention scores. We hope with advancements in Deep learning models and collecting more data, we can solve this problem in the future. We also think some other data formats, such as FMRI, are worth looking into for future work.

5 | Vessele Tree Topology

We studied vascular and non-vascular fundus image analysis. In chapter 2 3, we showed our result in the no-vascular analysis of the fundus images. In this chapter, we present our study in the vascular part of the fundus images.

5.1 Introduction

Retinal fundus images allow ophthalmologists to diagnose a variety of ocular and cardiovascular diseases, including diabetic retinopathy (DR), glaucoma, age-related macular Degeneration (AMD), and the likelihood of stroke, among others Abràmoff et al. (2010); Baker et al. (2008); Ferris et al. (2005); Kipli et al. (2018); Klonoff and Schwartz (2000). These diagnoses are based on tell-tale features in the fundus image that are correlated with a higher likelihood of certain diseases. For example, a higher cup-to-disk ratio in the optic nerve is correlated with a higher likelihood of glaucoma Bock et al. (2010). Similarly, the artery-vein ratio can be used to predict a patient's risk for diabetic retinopathy, and hypertension Abràmoff et al. (2010); Ikram et al. (2013); Pakter et al. (2001,0). Other diagnostically useful features include vessel tortuosity, bifurcation, branching angles, the presence of exudates, etc. Abràmoff et al. (2010); Claudia Kondermann (2007); Hart et al. (1999); Ikram et al. (2004); Joshi and Karule (2018); Lu et al. (2016); Rochtchina et al. (2007).

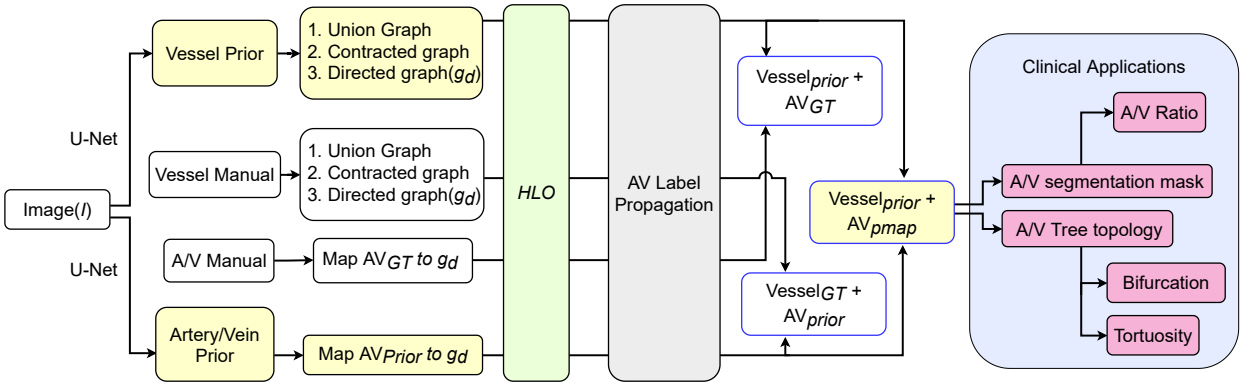

Figure 5.1: Flowchart of the whole pipeline: We start from a fundus image, then we generate Vessel Probability map(Vessel prior, fig. 5.2.c), Artery/Vein Probability Map(AV Prior, fig. 5.2.e) from two separately trained U-Net CNNs, we use multilevel skeletonization to produce union graph of the vasculature which is further pruned significantly with a unique graph contraction technique. Then we use parallel Dijkstra's shortest path algorithm to extract an un-directed topology graph, which is assigned direction by our Flow assignment algorithm to obtain a directed graph(g_d). We then map the AV_{prior} labels to g_d. Then we perform a series of High-Level Graph operations (**HLO**) to minimize topology cost function Eq. 5.1. We then perform a simple A/V label propagation algorithm to show that our topology graph is correct using different measures. **Ablation**: We perform heavy ablation studies(shown by blocks in light blue color) by replacing vessel prior with manual vessel segmentation and AV prior by AV manual segmentation, alternating one another such that at a given point, we have at least one prior in the pipeline. The pipeline works best when we have good priors(Vessel segmentation and AV label priors). As such, we can always improve our priors and come back to obtain better performance.

.

Medical image analysis tasks mostly fall into classification and segmentation tasks as we need to identify some regions with some abnormality to detect diseases. As such, we use a Fully Connected Convolutional Neural Network (FCNN) to perform the segmentation tasks. We assign a class label (0, 1, ..., k) to each pixel in the segmentation task. U-Net, a multi-level encoder-decoder-based architecture is used as a backbone for major medical image segmentation tasks Ronneberger et al. (2015). It has a bottleneck part on each level that helps ignore the noise and focus on information that matters, also allowing the skip connection to keep any information lost in the process. It uses a sliding window approach to make such architecture work with any size images, so we have used U-Net as the backbone for

our intermediate feature extraction tasks as in figure 5.1, where we use two U-Nets for vessel probability-map assignment and Artery/vein probability-map assignment. Usually, we use a threshold to binarize such probability maps to obtain a vessel mask. Using a fixed mask can lead to a very annoying trade-off problem of precision and recall: a very low threshold captures too much noise, whereas a high threshold misses a lot of fainter vessels. More importantly, only a binary mask is not enough to measure features like tortuosity, branching factors, vessel crossing anomaly, Artery/Vein ratio, and more. Thus in this research, we aim to extract the entire vasculature as a graph topology such that we can traverse to any part of the vascular structure and take all sorts of before-mentioned measurements, also run complex structural analysis on it Ikram et al. (2013); Patton et al. (2006).

In order to capture the complete vasculature, we use a novel idea of *multilevel-skeletonization* (5.3.1) where we use a set of thresholds ranging from low to high and capture the complete vascular structure. Then, we convert the combined skeleton generated from binary images of each threshold and convert that into a lattice graph called *union-graph*. In this union graph, the cyan-colored nodes are *branch-nodes*($degree > 2$), and golden nodes are called *end-nodes*($degree == 1$). Naturally, the branch nodes should only appear in the branching locations. However, we can see the union graph being dense. Thus, we use a novel pruning technique called union-graph-contraction to get rid of the significant amount of nodes and edges, keeping the overall vascular structure intact. Then we use 2-step Dijkstra's shortest path algorithm to extract the clean graph structure of the vasculature. We then assign direction**??** on this contracted graph along with the concept of Pseudo Optic Nerve Head Assignment to detect disconnected regions. The extracted directed topology graph opens up numerous potential research paths for downstream analysis and diagnosis tasks. For

instance, in this research, we used the topology to propagate Artery/Vein label to obtain a better A/V segmentation mask, which is used to calculate a crucial bio-marker called Artery/Vein ratio that reveals arteriolar narrowing like to hypertensive retinopathy, stroke and coronary artery disease Baker et al. (2008); Pakter et al. (2005). Similarly, one can easily run analyses like branching factor analysis, tortuosity, crossing anomaly, and so on.

We used two priors to do A/V label propagation; A/V probability obtained from U-Net, and directed graph topology obtained by the flow-assignment algorithm. There are some inherent limitations stemming from the skeletonization algorithm in the union skeleton algorithm. That shifts some crossing either upward or downward, introducing unnecessary **sink** and **source** nodes as shown in white-colored nodes. Then we do a handful of simple high-level operations(HLO) that we can perform on the directed topology graph to fix such problems. We simply minimize a vessel topology cost function by performing a series of such operations as mentioned. Once we have a low-cost topology graph, we simply propagate Artery/Vein labels by using the simple propagation strategy. We also did very crucial ablation studies that reflect on how the quality of priors (Vessel probability and Artery/Vein) affects the overall quality of the topology and how it affects the downstream tasks like A/V label propagation.

5.2 Related Work

Fundus images are widely used for vesicular and eye-related disease detection. A good vascular topology can lead to more accurate disease detection. Over the years, many researchers tried different approaches to tackle this challenging problem. Some researchers applied graph-based approaches to solve this problem Amil et al. (2019), Zhao et al. (2020),

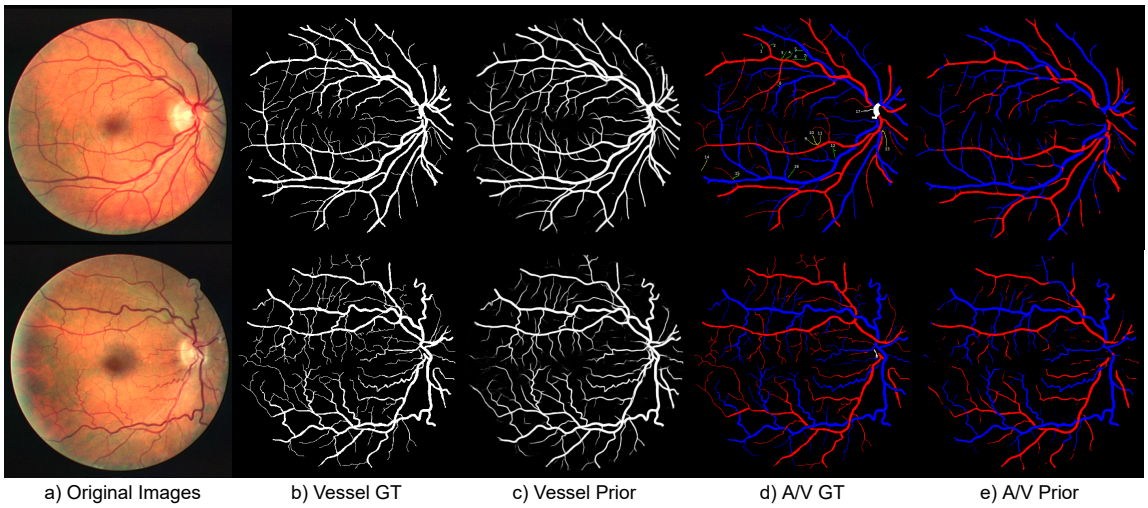

Figure 5.2: a). Original images b). Manual vessel segmentation c). Vessel prior obtained from U-Net d). AV prior generated from U-Net

.

Estrada et al. (2015a). In recent years many researchers applied deep learning techniques for both segmentation Roychowdhury et al. (2015), Hemelings et al. (2019), Girard et al. (2019) and classifications Xu et al. (2017), Lv et al. (2020), Hu et al. (2021) of fundus images.

5.2.1 *Graph topology*

In Amil et al. (2019), Amil et al. applied the structure of the blood vessels to detect patients with diabetic retinopathy and glaucoma. First, use an automatic unsupervised segmentation algorithm to extract the tree-like graph. Then quantify structural differences between the graph of healthy and non-healthy patients. Fractal analysis was used to characterize the extracted graphs. The paper claimed there is a significant difference between healthy and mom-healthy images. Zhao et al. in Zhao et al. (2020) used a dominant set clustering and formulated the retinal blood vessel topology estimation and A/V classification in the pairwise clustering system. Image segmentation, skeletonization, and significant node identification are used for graph construction. Also, the inverted Euclidean distance is applied between

two endpoints. Classification into A/V is applied based on intensity and morphology after constructing the vascular network. In the previous work, Estrada et al. in Estrada et al. (2015a) applied a graph-theoretic framework to differentiate arteries from veins. The use of underlying vessel topology made a better classification for small and midsize vessels. Applying the global likelihood model explores the possible solutions relevant to project vessels. The model is capable of analyzing the entire vasculature. The current proposed model is an extension and fully automated version of this prior work.

5.2.2 *Segmentation*

Roychowdhury et al. in Roychowdhury et al. (2015) applied a three-stage algorithm for blood vessel segmentation. In the first stage, a binary image is extracted from the high-pass filtering on the green plane of the fundus image. Also, another binary image of the vessel region was extracted from the morphologically reconstructed enhanced image. If the region is common in both binary images, it will be extracted as a major vessel. In the second step, by using a Gaussian mixture model (GMM), the remaining pixels in two binary images are classified. The classifier uses a set of eight features, which are extracted based on the pixel neighborhood, first, and second-order gradient images. The major portions of the blood vessels and the classified vessel pixels are combined in the third step. Roychowdhury et al. claim less dependent on training data, less segmentation time, and consistent vessel segmentation accuracy on normal images in their algorithm. Hemelings et al. in Hemelings et al. (2019) applied deep learning architecture for A/V segmentation. To be more specific U-net architecture is applied for segmentation. The model was tested on DRIVE and HRF datasets. Girard et al. combined deep learning and graph propagation in Girard et al.

(2019) for A/V segmentation. The convolutional neural network (CNN) is used to segment and classify vessels into arteries and veins jointly. After the initial CNN labeling, a graph theory is used to calculate the cost of linking the pairs of branches that represent vessels. A minimum spanning tree is used to simplify the graph and improve efficiency.

5.2.3 *Classification*

Xu et al. used an automated computer-aided method to predict A/V classification. In order to reduce the differences in feature space, Intra-image regularization, and inter-subject normalization were applied. This method was tested on DRIVE DatasetXu et al. (2017). Attention-guided U-net with atrous convolution(AA-UNet) applied by Lv et al. (2020) to segment the vessel and non-vessel pixels. They use AA-UNet to generate the attention mask and applied as a weighting function to increase the network attention toward the vessel region. Atrous convolution instead of ordinary convolution in the feature layer increases the receptive field and decreases computation. In Hu et al. (2021), Hu et al. applied a vessel-constraint network(VC-Net) to enhance A/V classification. This method utilizes the information on vessel distribution and edge. VC-Net uses a vessel-constraint (VC) module, which combines local and global vessel information. The model generates a weight map to constrain the A/V features, which enhance the edge and end features of blood vessels. It also suppresses the background-prone features. In order to extract the blood vessel information and improve the feature extraction capability and robustness of the model, a multiscale feature (MSF) module was used. The VC-Net get vessel segmentation result simultaneously.

5.3 Methodology

In this section, we describe the three major steps of our fully automated vessel extraction method: (1) topology extraction with multilevel skeletonization, (2) topology estimation, and (3) artery-vein label propagation.

5.3.1 *Topology Extraction with Multilevel Skeletonization*

The first step is to extract vascular topology and measure different important factors like (Artery to Vein)AV-ratio, tortuosity, and bifurcation mentioned in the section 5.1. It is crucial to segment vasculature correctly in order to compute usable measures in the original image. Thus, our method works on the original image size without having to resize. Funduscope, an imaging device to capture fundus images, comes in different sizes and quality and can have different resolution images. We have shown that our technique works in a variety of datasets end-to-end with no minimum parameter optimization.

Extraction of Likelihood-Map

We use Convolutional Neural Network on the green channel to get the vessel likelihood map. Specifically, we use a U-Net architecture Ronneberger et al. (2015) with stochastically weighted loss Khanal and Estrada (2020b) to train k-fold for DRIVE Staal et al. (2004) and WIDE Estrada et al. (2015b). The stochasticity helps detect finer vessels with easeKhanal and Estrada (2020b).

Extraction of AV-Prior

We extract the likelihood for Artery/Vein for each pixel by k-fold cross-validation with U-Net. Such pixel-wise classification is inconsistent to actually use for measures like AV ratio and bifurcation because of too many misclassifications and noise as seen in figure 5.2 e. Thus, we use it as a prior to topology optimization and AV classification.

Multilevel Skeletonization

We then use a range of thresholds given by T on $\hat{I}_p$, which ranges from $255 - t_0$ with a step size of p to extract multilevel-skeleton. This skeleton is converted to a dense lattice graph called a union graph. We further used a novel pruning technique to contract the graph nodes/edges to reduce the size significantly.

Dijkstra shortest path algorithm for complete vessel path tracing

We use Parallel Dijkstra Shortest path algorithm to estimate the vessel path on the graph G_{uc}. Being lightweight and parallel, the running time is greatly reduced. We use the geometric mean of the node attributes mentioned in section 5.3.1 to weight the edges as below:

Tree topology cost function.

We use an intuitive cost function to optimize the topology:

$$C_{g_d} = \sum_{b \in B_g} e^{CR_{pair}(b)} + e^{CR_{prob}(b)} + e^{BR(b)} + e^{FW(b)} \tag{5.1}$$

, where g is the un-directed graph of g_d, CR_p is crossings cost that looks if illegal crossings are made, CR restricts the correct distribution of A/V in branches, BR is a branchless cost that respects the true nature of vessel branching, and FW cost keeps the blood flow between A-A/V-V in check.

5.4 Discussion

Different diseases have different pathological markers uniquely identified in a fundus image. Most Deep learning techniques tend to only diagnose the absence or presence of a single disease. This leads to extra time, effort, and extra money to make another system to diagnose another disease. In this work, we attempt to move forward to the first step to extract all the vascular features of fundus images (fig. 5.1). Features like Artery/Vein ratio, tortuously, bifurcation, and disconnected vessels can be extracted following this work as we have shown that we have extracted both artery/Vein mask and their topology. This work is a pipeline toward it. We have shown that its performance of it depends on the quality of the priors we feed to the system. Furthermore, we have shown that with decent priors, the algorithm trained on the DRIVE dataset works well with another dataset with no minimum parameter calibration. We are confident that this work enables a solid/more principled path for automated fundus image diagnosis. In addition, we can see how our flow-assignment algorithm detects possible points where vessels are broken. The size of the nodes depicts if a natural blood path has been broken or obstructed. For example, we show the disconnected vessel endpoint gets more visits than the other ones. We have only used the directed graph obtained from the flow-assignment algorithm. However, an intuitive depiction of blood flow obstruction could be used in future research as an important factor in itself.

5.5 Conclusion and Future Work

Automated diagnosis of diseases is not fully trusted in the medical field. One reason is that is not explainable enough. Thus we aimed to enable the diagnosis of fundus image by extracting most of the features like a human.

6 | CONCLUSION

This thesis is focused on feature detection of the non-vascular part of the fundus image, and that knowledge can be used for non-vascular disease detection. We are very confident that such feature extraction can boost the performance of disease detection models 2 3. We put in extensive work on vessel segmentation which is very critical in vessel disease detection 5. We also believe researchers can benefit from both vascular and non-vascular feature detection to boost the performance of their model. Moreover, our work in MRI-3d images can be a good foundation for future researchers. We hope with the advancement of Deep learning models and techniques. Researchers can benefit from the work that we have done in this area of research and advance the attention score prediction and help people with ADHD in the future 4

References

Abdullah, A. S., Özok, Y. E., and Rahebi, J. (2018). A novel method for retinal optic disc detection using bat meta-heuristic algorithm. *Medical & Biological Engineering & Computing*, 56(11):2015–2024.

Abràmoff, M. D., Garvin, M. K., and Sonka, M. (2010). Retinal imaging and image analysis. *IEEE Rev Biomed Eng*, 3:169–208.

Amil, P., Reyes-Manzano, C. F., Guzmán-Vargas, L., Sendiña-Nadal, I., and Masoller, C. (2019). Network-based features for retinal fundus vessel structure analysis. *PloS one*, 14(7):e0220132–e0220132. 31344132[pmid].

Baker, M. L., Hand, P. J., Wang, J. J., and Wong, T. Y. (2008). Retinal signs and stroke: revisiting the link between the eye and brain. *Stroke*, 39(4):1371–1379.

Bock, R., Meier, J., Nyúl, L. G., Hornegger, J., and Michelson, G. (2010). Glaucoma risk index:automated glaucoma detection from color fundus images. *Medical Image Analysis*, 14(3):471–481.

Carmona, E. J., Rincón, M., García-Feijoó, J., and Martínez-De-La-Casa, J. M. (2008). Identification of the optic nerve head with genetic algorithms. *Artificial Intelligence in Medicine*, 43(3):243–259.

Cheng, J., Liu, J., Wong, D. W. K., Yin, F., Cheung, C., Baskaran, M., Aung, T., and Wong, T. Y. (2011). Automatic optic disc segmentation with peripapillary atrophy elimination. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 6224–6227.

Claudia Kondermann, Daniel Kondermann, M. Y. (2007). Blood vessel classification into arteries and veins in retinal images. *Proc. SPIE*, 6512.

Digre, K. B., Nakamoto, B. K., Warner, J. E., Langeberg, W. J., Baggaley, S. K., and Katz, B. J. (2009). A comparison of idiopathic intracranial hypertension with and without papilledema. *Headache: The Journal of Head and Face Pain*, 49(2):185–193.

Estrada, R., Allingham, M. J., Mettu, P. S., Cousins, S. W., Tomasi, C., and Farsiu, S. (2015a). Retinal artery-vein classification via topology estimation. *IEEE Transactions on Medical Imaging*, 34(12):2518–2534.

Estrada, R., Tomasi, C., Schmidler, S. C., and Farsiu, S. (2015b). Tree topology estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(8):1688–1701.

Ferguson, M., ak, R., Lee, Y.-T., and Law, K. (2017). Automatic localization of casting defects with convolutional neural networks. pages 1726–1735.

Ferris, F. L., Davis, M. D., Clemons, T. E., Lee, L.-Y., Chew, E. Y., Lindblad, A. S., Milton, R. C., Bressler, S. B., Klein, R., and Group, A.-R. E. D. S. A. R. (2005). A simplified severity scale for age-related macular degeneration: Areds report no. 18. *Archives of ophthalmology (Chicago, Ill. : 1960)*, 123(11):1570–1574. 16286620[pmid].

Fraz, M. M., Remagnino, P., Hoppe, A., Uyyanonvara, B., Rudnicka, A. R., Owen, C. G., and Barman, S. A. (2012). An ensemble classification-based approach applied to retinal blood vessel segmentation. *IEEE Transactions on Biomedical Engineering*, 59(9):2538–2548.

Girard, F., Kavalec, C., and Cheriet, F. (2019). Joint segmentation and classification of retinal arteries/veins from fundus images. *Artificial Intelligence in Medicine*, 94:96–109.

Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., and Schölkopf, B. (2009). *Covariate shift and local learning by distribution matching*, pages 131–160. MIT Press, Cambridge, MA, USA.

Harizman, N., Oliveira, C., Chiang, A., Tello, C., Marmor, M., Ritch, R., and Liebmann, J. M. (2006). The ISNT Rule and Differentiation of Normal From Glaucomatous Eyes. *Archives of Ophthalmology*, 124(11):1579–1583.

Hart, W. E., Goldbaum, M., Côté, B., Kube, P., and Nelson, M. R. (1999). Measurement and classification of retinal vascular tortuosity. *International Journal of Medical Informatics*, 53(2):239–252.

Hemelings, R., Elen, B., Stalmans, I., Van Keer, K., De Boever, P., and Blaschko, M. B. (2019). Artery–vein segmentation in fundus images using a fully convolutional network. *Computerized Medical Imaging and Graphics*, 76:101636.

Hu, J., Wang, H., Cao, Z., Wu, G., Jonas, J. B., Wang, Y. X., and Zhang, J. (2021). Automatic artery/vein classification using a vessel-constraint network for multicenter fundus images. *Frontiers in cell and developmental biology*, 9:659941–659941. 34178986[pmid].

Ikram, M. K., Cheung, C. Y., Lorenzi, M., Klein, R., Jones, T. L. Z., and Wong, T. Y. (2013). Retinal vascular caliber as a biomarker for diabetes microvascular complications. *Diabetes Care*, 36(3):750–759.

Ikram, M. K., de Jong, F. J., Vingerling, J. R., Witteman, J. C. M., Hofman, A., Breteler, M. M. B., and de Jong, P. T. V. M. (2004). Are Retinal Arteriolar or Venular Diameters Associated with Markers for Cardiovascular Disorders? The Rotterdam Study. *Investigative Ophthalmology & Visual Science*, 45(7):2129–2134.

Joshi, S. and Karule, P. (2018). A review on exudates detection methods for diabetic retinopathy. *Biomedicine & Pharmacotherapy*, 97:1454–1460.

Joshua, A. O., Nelwamondo, F. V., and Mabuza-Hocquet, G. (2019). Segmentation of optic cup and disc for diagnosis of glaucoma on retinal fundus images. *2019 Southern African Universities Power Engineering Conference/Robotics and Mechatronics/Pattern Recognition Association of South Africa (SAUPEC/RobMech/PRASA)*, pages 183–187.

Khanal, A. and Estrada, R. (2020a). Dynamic deep networks for retinal vessel segmentation. *Frontiers in Computer Science*, 2:35.

Khanal, A. and Estrada, R. (2020b). Dynamic deep networks for retinal vessel segmentation. *Frontiers in Computer Science*, 2:35.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.

Kipli, K., Hoque, M. E., Lim, L. T., Mahmood, M. H., Sahari, S. K., Sapawi, R., Rajaee, N., and Joseph, A. (2018). A review on the extraction of quantitative retinal microvascular image feature. *Computational and mathematical methods in medicine*, 2018:4019538–4019538. 30065780[pmid].

Klonoff, D. C. and Schwartz, D. M. (2000). An economic analysis of interventions for diabetes. *Diabetes Care*, 23(3):390–404.

Lu, Y., Serpas, L., Genter, P., Mehranbod, C., Campa, D., and Ipp, E. (2016). Disparities in diabetic retinopathy screening rates within minority populations: Differences in reported screening rates among african american and hispanic patients. *Diabetes Care*, 39(3):e31–e32.

Lv, Y., Ma, H., Li, J., and Liu, S. (2020). Attention guided u-net with atrous convolution for accurate retinal vessels segmentation. *IEEE Access*, 8:32826–32839.

Mangipudi, P. S., Pandey, H. M., and Choudhary, A. (2021). Improved optic disc and cup segmentation in glaucomatic images using deep learning architecture. *Multimedia Tools and Applications*, 80(20):30143–30163.

Maninis, K.-K., Pont-Tuset, J., Arbeláez, P., and Gool, L. V. (2016). Deep retinal image understanding. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*.

Mohan, D., Harish Kumar, J. R., and Sekhar Seelamantula, C. (2018). High-performance optic disc segmentation using convolutional neural networks. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 4038–4042.

nih (1999). ABCD Data Repository abcd description.

Pakter, H., Fuchs, S., Maestri, M., and Fuchs, F. (2001). Detection of hypertensive retinopathy findings: a study of agreement between ophthalmologist and cardiologist. In *INVESTIGATIVE OPHTHALMOLOGY & VISUAL SCIENCE*, volume 42, pages S699–S699. ASSOC RESEARCH VISION OPHTHALMOLOGY INC 9650 ROCKVILLE

PIKE, BETHESDA, MD ....

Pakter, H. M., Ferlin, E., Fuchs, S. C., Maestri, M. K., Moraes, R. S., Nunes, G., Moreira, L. B., Gus, M., and Fuchs, F. D. (2005). Measuring arteriolar-to-venous ratio in retinal photography of patients with hypertension: Development and application of a new semi-automated method. *American Journal of Hypertension*, 18(3):417–421.

Pallawala, P. M. D. S., Hsu, W., Lee, M. L., and Eong, K.-G. A. (2004). Automated optic disc localization and contour detection using ellipse fitting and wavelet transform. In Pajdla, T. and Matas, J., editors, *Computer Vision - ECCV 2004*, pages 139–151, Berlin, Heidelberg. Springer Berlin Heidelberg.

Patton, N., Aslam, T. M., MacGillivray, T., Deary, I. J., Dhillon, B., Eikelboom, R. H., Yogesan, K., and Constable, I. J. (2006). Retinal image analysis: Concepts, applications and potential. *Progress in Retinal and Eye Research*, 25(1):99–127.

Rochtchina, E., Burlutsky, G., Liew, G., Mitchell, P., Wang, J. J., Klein, B. E., Knudtson, M. D., Klein, R., and Wong, T. Y. (2007). Retinal vessel diameter and cardiovascular mortality: pooled data analysis from two older populations. *European Heart Journal*, 28(16):1984–1992.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*.

Roychowdhury, S., Koozekanani, D. D., Kuchinka, S. N., and Parhi, K. K. (2016). Optic disc boundary and vessel origin segmentation of fundus images. *IEEE Journal of Biomedical and Health Informatics*, 20(6):1562–1574.

Roychowdhury, S., Koozekanani, D. D., and Parhi, K. K. (2015). Blood vessel segmentation of fundus images by major vessel extraction and subimage classification. *IEEE Journal of Biomedical and Health Informatics*, 19(3):1118–1128.

Shen, C., Luo, Q., Chamberlain, S. R., Morgan, S., Romero-Garcia, R., Du, J., Zhao, X., Évelyne Touchette, Montplaisir, J., Vitaro, F., Boivin, M., Tremblay, R. E., Zhao, X.-M., Robaey, P., Feng, J., and Sahakian, B. J. (2020). What is the link between attention-deficit/hyperactivity disorder and sleep disturbance? a multimodal examination of longitudinal relationships and brain structure using large-scale population-based cohorts. *Biological Psychiatry*, 88(6):459–469. Attention-Deficit Disorder and Autism Spectrum Disorder.

Sivaswamy, J., Krishnadas, S. R., Datt Joshi, G., Jain, M., and Syed Tabish, A. U. (2014). Drishti-gs: Retinal image dataset for optic nerve head(onh) segmentation. In *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*, pages 53–56.

Staal, J., Abramoff, M., Niemeijer, M., Viergever, M., and van Ginneken, B. (2004). Ridge based vessel segmentation in color images of the retina. *IEEE Transactions on Medical Imaging*, 23(4):501–509.

Varma, R., Steinmann, W. C., and Scott, I. U. (1992). Expert agreement in evaluating the optic disc for glaucoma. *Ophthalmology*, 99(2):215 – 221.

Walter, T. and Klein, J.-C. (2001). Segmentation of color fundus images of the human retina: Detection of the optic disc and the vascular tree using morphological techniques. In *ISMDA*.

Wang, C.-L., Hsieh, M.-Y., Hung, Y.-W., Tsai, M.-H., Chan, M.-H., Chen, J.-M., and Tung, K.-C. (2016). Retina image–based optic disc segmentation. *Advances in Mechanical Engineering*, 8(6):1687814016649298.

Xu, X., Ding, W., Abràmoff, M. D., and Cao, R. (2017). An improved arteriovenous classification method for the early diagnostics of various diseases in retinal image. *Computer Methods and Programs in Biomedicine*, 141:3–9.

Youssif, A. A. A., Ghalwash, A. Z., and Ghoneim, A. A. S. A. (2008). Optic disc detection from normalized digital fundus images by means of a vessels' direction matched filter. *IEEE Transactions on Medical Imaging*, 27(1):11–18.

Zhao, Y., Xie, J., Zhang, H., Zheng, Y., Zhao, Y., Qi, H., Zhao, Y., Su, P., Liu, J., and Liu, Y. (2020). Retinal vascular network topology reconstruction and artery/vein classification via dominant set clustering. *IEEE Transactions on Medical Imaging*, 39(2):341–356.

Zhu, X., Rangayyan, R. M., and Ells, A. L. (2010). Detection of the optic nerve head in fundus images of the retina using the hough transform for circles. *Journal of digital imaging*, 23(3):332–341. 19238486[pmid].